# 地震学中的计算方法

朱介寿　等编著

地 震 出 版 社

# 序　言

近年来，由于地震学各领域广泛使用了计算机，使得这方面的数值方法及计算机程序有很大发展。例如，有关地震信号的存储与处理问题，研究震源及地球内部结构问题，地震过程的模拟问题，都发表了一系列重要的论文及专著。国内外也多次召开地震学计算问题的专业会议。

本书从实际应用出发，对地震学中的一些基本问题，就其数学模型、数值计算方法及计算程序作了较详尽的讨论。

第一章至第三章对地球物理反演理论（特别是线性反演理论）作了详尽的论述。并着重讨论了解反演问题的基本数值方法，其中包括广义逆方法、无约束最优化方法及线性非线性规划等。前三章是本书后继各章处理反演问题的基础。

第四章讨论P波初动断层面解的计算机处理方法。从简单的单力偶或双力偶模式出发，根据概率统计原理，得到了用计算机搜寻断层面及有关震源运动参数的数值计算方法。

第五章讨论震源基本参数（震中位置、震源深度及发震时刻）的计算机定位方法。对于近震，分别叙述了初定震源参数及用Geiger法、Marquardt法、Powell法修定的算法。对于远震，讨论了用Geiger法修定震源参数的原理及数值计算方法。

第六章讨论地震体波在水平及倾斜层状介质中传播的正演及反演问题解法。这些介质可以是均匀各向同性的，也可以是沿垂直方向速度呈线性变化的。

第七章叙述了用Herglotz-Wiechert公式反演地球内部速度分布的基本算法及改进方法。给出了各种速度分布的球层模型的理论走时计算公式。

第八章详细讨论了用面波研究地壳地幔结构的原理及方法。对于计算水平层状介质面波理论频散的矩阵法及快速递推算法作了详尽推导，并仔细叙述了用地震面波资料进行计算机处理及数值反演的方法。

第九章详细论述了计算理论地震图的反射率法（这是目前精度较高、应用较广的方法）的原理及数学推导，并对数值计算方法及程序设计中有关问题作了讨论，给出了计算实例。

第十章研究了任意速度分布及弯曲界面的介质中地震射线追踪问题。讨论了反射波及折射波通过介质的射线分布及走时的数值计算方法。同时还考虑了反演问题解法。

以上各章均有用FORTRAN 语言编写的计算机程序 及 数 值计算实例。

本书大部分内容是作者在前几年科研及教学工作中逐渐写成的。其中部分章节是以下同志写的：周家纪（第九章），彭君维（第十章），左海燕（第三章部分）。

在本书写作过程中，作者得到了张之立、丁韫玉、冯锐、张少泉、钟本善、曹家敏、熊辉、张平、张雁如、阎志德等同志的大力支持和帮助。他们中间有的为本书提供了某些计算机程序，有的则提供了部分素材及图表。作者对上述同志表示 衷 心 的 感谢。

作者还得到了曾融生先生的热情指导。傅承义先生及王仁先生亦对作者给予多方面的关怀和鼓励。在此，谨向上述各位先生致以深切的谢意。

本书原稿虽经几次修改及补充，限于作者水平，难免有疏漏不妥之处，敬请读者批评指正。

# 目　录

# 第一章　地球物理反演问题概论

## §1-1　引　　言

目前，有关地球内部，特别是它的深部的知识，绝大部分来源于对地表地球物理观测资料的解释。因此，任何一种地球物理观测最终都要求解反演问题。即根据各种位场、地震波、地球自由振荡、交变电磁场、以及热学或光学的地球物理观测数据，去推测地球内部的结构形态及物质成分，定量计算各种有关的物理参数。

解反演问题是一门广泛应用于各个科学技术领域的基本研究方法。凡科学技术中涉及到观测与演绎的，都会碰到这类问题。例如，天体物理根据光谱的测定研究宇宙的物质组成及星体的演化；根据人造卫星瞬时位置及速度的观测数据计算其轨道要素；利用气象观测数据研究大气层运动规律；根据大地测量数据可以推算出近代应力场等。有人估计，约有75%的科学技术问题，都以不同方式，在不同程度上涉及到解反演问题。

解反演问题的古典方法是最小二乘法及统计学的回归、参数估计等。近廿余年来，由于地球物理反演问题的计算广泛应用了信息论、线性及非线性规划、广义逆理论及最优化方法等一些数学工具，在理论和方法上都有重大进展。

任何反演问题的计算都是建立在正演问题基础上的。只有我们已经了解到某种地球物理问题的数学解答(解析的或数值的)，即能计算其正演问题时，我们才可能解反演问题。对于那些成因或机制尚不清楚的地球物理问题，是谈不到解反演问题的。

反演问题是地球物理研究中一个重要而又困难的课题。最主要的困难是反演结果存在着多解性(不唯一性)，这就降低了反演结果的可靠程度。

地球物理反演问题的不唯一性不是反演方法或技巧上的缺陷引起的，而是这种问题所存在的固有困难。困难之一是观测资料不完备。这是由于地球物理观测通常只能在地球表面或其上空进行，无法获得反映地球内部物质的充足的信息。这就使得反演结果有多种可能性。而每一种可能的解答都能够满足不充分的观测数据。地球物理观测信息的不足是无法用数学技巧来弥补的。困难之二是任何地球物理观测都存在干扰和误差。用这种受畸变的观测数据进行反演计算是不稳定的，即观测数据中的少量错误可能导致反演结果很大的变动。因此，在地球物理反演理论中，不仅要讨论各种解算方法，还要着重研究如何减小解的不唯一性问题。目前，国内外地球物理工作者已经在这方面作了不少努力，并取得了一定的进展。

## §1-2 反演理论的发展

在地球物理学发展的初期，由于观测资料比较粗糙，一般只对测量数据作定性或粗略的定量解释。从本世纪初到五十年代，随着地球物理方法的发展，各种解反演问题的定量计算方法也随之提出来。但这些方法大多是孤立地为解决某一特定问题而提出的。例如，根据地球表面体波走时观测来反演速度随深度变化的Herglotz-Wiechert公式，就是基于假设速度是深度的单调函数时的一种积分运算。又如，重力、地磁等位场解释中常用到的特征点法、积分法，也是为解释某一孤立位场异常常用的一些简单运算方法。这些方法的特点是仅能提供有关平均化的参量，例如异常物体的质量或埋藏深度等。值得指出的是，这一阶段提出的基于理论计算曲线与观测曲线相拟合的选择法（如直流电测深

及大地电磁测深解释中应用的理论曲线量板)，已具有现代解释理论的萌芽，这种方法的基本原理目前仍在应用。应当指出，上述各种反演方法都假定了不存在误差，故又称精确计算法。

近三十余年来，随着科学技术的迅速发展，地球物理资料无论在空间分布范围上(海洋、大陆、大气层及太空)，以及数据的可靠性和丰富程度上，都较过去有很大改进。宇宙飞行器的发展，使我们取得了大量的月球、火星、金星、水星、木星及土星等物理数据。这就为研究地球内部以及行星内部的物质结构，特别是它们的精细结构，提供了较丰富的资料。数学方法及电子计算机的广泛应用，又促进了地球物理一般反演理论的建立和数据处理自动化的发展。

用计算机对地球物理资料进行反演计算，其基本方法是寻找某种模型，使此模型的理论计算值与观测值(存在误差的)在最小二乘意义下拟合。一般称为最小二乘拟合问题。在不同的科学技术领域内，反演问题的提法是不同的。例如，从数学观点来看，反演是在给定的子空间内找寻与函数空间中一点最靠近的点。从统计学观点来看，反演是回归或参数估计问题。而从信息论观点来看，反演又可看成滤波或过程鉴别问题。尽管术语或提法不同，但它们都表达同一内容，即从观测数据中反演计算所要求的地球内部物质的物理参数。

有关这类工作较早期的文献可参看Corbato(1965)，Johnson(1969)，Nage 及 Garde(1969)，Tanner(1967)等人文章，这些都是用最小二乘法在计算机上自动拟合进行反演的方法。

1967—1970年间，Backus及Gilbert发表了一系列重要文章。这些文章实际上奠定了近代地球物理线性反演理论的基础，具有重要意义。这个比较严格完整的反演理论以后被称为Backus-Gilbert 理论(或B-G理论)。在他们的原始文献中，一般是研究整个地球内物理参数的连续变化，故反演问题以连续型函数表达。1972年，Wiggins用矩阵将Backus-Gilbert理论表达

为离散形式。同年，Jackson亦用广义逆理论详细讨论了线性反演问题在各种情况下的解答。

Backus-Gilbert 线性反演理论提出后，立即在各个地球物理领域中得到广泛应用。例如地震学中体波或面波反演的文献可参看，Der，Masse及Landisman（1970），Knopoff（1972），Johnson及Gilbert（1972），Braile（1973），Braile及Keller（1975），Crosson（1976），Aki（1977）等人的文章。位场解释方面的文献可参见McGrath（1973），Braile，Keller及Peeples（1974），Oldenburg（1976），Parker（1975），William（1975），Vigneresse（1977，1978），Borsting Pedersen（1977），Green（1975）等。地球电磁场的反演文献可参见Parker（1970），Inmann，Ryu及Ward（1973），Inmann（1975）等。

前已指出，地球物理反演结果是不唯一的。要减少这种多解性，主要的办法是进行约束，即尽可能利用各种已知的资料（先验信息）对解答范围进行限制。因此，近几年内，地球物理反演理论已逐渐集中研究约束条件下的最优化问题。常用到的数学工具是线性或非线性规划等。例如，在Al-Chalabi（1972），Safon，Vasseur及Cuer（1977），Sabatier（1977），Vigneresse（1978），Kennett（1978），Jackson（1979），Fisher及Howard（1980）等人的文章中都集中讨论了这些问题的解法。

在线性反演理论中，普遍引入了分辨矩阵，信息矩阵等重要概念，这对于理解反演的实质及评价反演的效果有重要作用。这方面的内容可参看Backus及Gilbert的原著，Jackson（1972），Jupp（1975），Crosson（1976），Borsting Pedersen（1977）等人的文章。

以上，我们简要回顾了精确的及线性的反演理论的发展概况。与此并行的，解反演问题还可以用一种统计试验法（即蒙特卡罗法），其特点是通过计算机选择地球模型去拟合观测资料。这方面的工作可参看Wiggins（1969），Andersen，Worthington

及 Cleary (1972), Press (1968, 1970)等。特别是Press用大型计算机对多种地球物理资料进行拟合，从几百万个随机试验值中挑选出几个较好的地球模型，已成为一项著名的研究工作。

下面，我们将讨论地球物理反演理论的一般数学原理。设$\mathscr{M}$为地球物理参量值构成的集合，称为“地球模型”，$\mathscr{E}$为映射地球参数与物理观测场之间的算子，称为“地球泛函”，而$\mathscr{D}$则是地球物理观测值构成的集合，称为“地球数据”。正演问题就是地球模型$\mathscr{W}\in\mathscr{M}$得到观测数据映象。$d=\mathscr{E}(\mathscr{W})\in\mathscr{D}$，反过来，由由映象$d\in\mathscr{D}$获得所有可能的元素$\mathscr{W}\in\mathscr{M}$则称为反演问题，

如果“地球模型”集合$\mathscr{M}$及“地球资料”集合$\mathscr{D}$都是线性空间，且“地球泛函”$\mathscr{E}$是线性映射算子，则这样的反演问题称为线性的。可是，大部分的地球物理反演问题都不是线性的。为此，必须采用两种办法使其线性化。第一种办法是将“地球泛函”算子$\mathscr{E}$用线性算子$\mathscr{E}_0$代替。最常见的办法是在某一初始地球模型$\mathscr{W}_0$附近展开，使线性函数$\mathscr{E}_0(\mathscr{W})$与$\mathscr{E}(\mathscr{W})$充分接近。另一种办法是通过约束条件使得集合$\mathscr{M}$及$\mathscr{D}$均成为线性空间。

我们假定“地球模型”$\mathscr{M}$为希尔伯特空间$L_2(\Omega)$,其中$\Omega$是$R$空间中的有限域。通过$m$个“地球泛函”的核$G_i(r)\in L_2(\Omega)$。可以把模型空间$\mathscr{M}$映射到资料空间$\mathscr{D}$，由下述积分方程表示，

$$\int_{\Omega} G_i(r)\mathscr{W}(r)dr=g_i(\mathscr{W}),(i=1,2,\cdots,m). \tag{1-2.1}$$

式中$\mathscr{W}(r)$为连续型地球模型参数($r$可以看成地球半径),$g_i(\mathscr{W})$等于或近似等于观测资料$d_i$(存在估计误差)。由这样表达的问题是连续型的。

然而，如将域$\Omega$分为$n$几个子域$\Omega_1$，$\Omega_2$，…，$\Omega_n$(例如将地球内部划分为若干层)，使$\Sigma_i\Omega_i=\Omega$，则上述连续型函数就被离散化。地球模型参数被离散为$\mathscr{W}_k(k=1,2,\cdots,n)$，积分方程

(1-2.1)可以写为矩阵方程形式，

$$(G\mathscr{W})_i=\sum_{k=1}^{n}G_{ik}\mathscr{W}_k=g_i(\mathscr{W}),\qquad (i=1,2,\cdots,m)\tag{1-2.2}$$

式中，$G$为$m\times n$矩阵，模型参数$\mathscr{W}$为$n$维向量，观测数据$g$为$m$维向量。式(1-2.2)表达的问题是离散型的。

所谓线性反演问题就是从$m$个观测数据$d_1$，$d_2$，…，$d_m$中获得我们所要求的地球模型$\mathscr{W}$（或$n$个离散值$\mathscr{W}_1$，$\mathscr{W}_2$，…，$\mathscr{W}_n$）的信息。Backus及Gilbert提出的反演理论，就是对连续型函数形式(1-2.1)给出平均解的一种途径。而广义逆理论就是对离散型函数(1-2.2)式求解的基本方法。

## §1-3 连续型线性反演理论概述

前面已经指出，地球物理反演问题实质上是寻求一个物理模型来拟合观测值。但这往往是不唯一的。Backus 及Gilbert 主要改进在于从全部可能模型的希尔伯特空间，特别是从那些满足观测数的模型空间中减少了不唯一性。为此，可以根据模型的性质对它进行某些约束。例如模型数据的非负性质，反演计算模型与初始模型在参数空间的最短“距离”性质等，都可用来约束模型，使反演结果的不唯一性得到改进。

按照Backus及Gilbert的观点，地球模型不过是适应于特定地球物理问题的一种数学抽象。例如，对地震学而言，地球模型可以看为四个参数（密度$\rho$，体积模量$\kappa$，剪切模量$\mu$，品质因数$Q$）随深度变化的函数。在地球化学中，地球模型将是各种元素的富集及组合情况随深度及时间变化的函数。在地球电磁场中，地球模型可以模拟为电导率随深度变化的函数。由此可见，对于大多数地球物理问题来说，地球模型可以表达为地球半径$r$的函数$\mathscr{W}(r)$，其中$0\leqslant r\leqslant a$，$a$是地球半径，平均值为$6.37\times10^6$米。

$\mathscr{W}$ 也可以视为希尔伯特空间中的一个元素。显然，对于 $\mathscr{W}$ 为负的那些子集，是没有地球物理意义的，在反演中应予排除。

地球物理反演问题的一般提法是，根据一组观测值 $E_j$，$j=1,2,\cdots,m$，找出适应于该观测值的地球模型 $\mathscr{W}(r)$。一般说来，观测值 $E_j$ 是模型 $\mathscr{W}$ 的非性线函数。

$$E_j = E_j(\mathscr{W}),\ j=1,\ 2,\cdots,\ m. \tag{1-3.1}$$

对于真实地球模型 $\mathscr{W}_0$，其相应的观测值为

$$E_j^0 = E(\mathscr{W}_0),\ j=1,\ 2,\cdots,\ m. \tag{1-3.2}$$

引入希尔伯特空间进行描述目的之一，是使我们能定义并计算观测数据相对于模型的导数。这可以考虑两个模型 $\mathscr{W}$ 及 $\mathscr{W}'$，它们之间仅相差一个微量 $\delta_m$，

$$\mathscr{W}' = \mathscr{W} + \delta_m. \tag{1-3.3}$$

若它们的函数值有关系

$$E_j(\mathscr{W} + \delta_m) = E_j(\mathscr{W}) + (F_j,\ \delta_m) + \varepsilon(\delta_m), \tag{1-3.4}$$

式中每一个 $F_j$ 都是希尔伯特空间的一个元素，$(F_j,\delta_m)$ 为 $F_j$ 与 $\delta_m$ 的内积，且 $\varepsilon(\delta_m)$ 比范数 $\|\delta_m\|$ 更快趋于零。如（1-3.4）式成立，则称此函数是 Fréchet 可微的（参见 Dunford 及 Schwartz，1958）。$F_j$〔或表示为函数 $F_j(r)$，$0\leqslant r\leqslant a$〕则称为观测数据 $E_j$ 的 Fréchet 核函数（或算子）。通常，$F_j$ 是 $E_j$ 对 $\mathscr{W}$ 的导数。所有地球物理有意义的数据都可以看成Fréchet可微的。显然，（1-3.4）式实质上是将问题线性化的一个重要步骤

设（1-3.3）式中的 $\mathscr{W}$ 是初始模型。而 $\mathscr{W}'$ 是真实模型，或至少是满足观测数据的模型。由于初始模型 $\mathscr{W}$ 是我们给定的，而 $\mathscr{W}'$ 是待求的，所以我们希望求得 $\delta_m$。但是 $\delta_m$，是无穷维向量空间的一个元素，而仅有 $m$ 个观测数据列出的 $\mathscr{W}$ 个观测方程，这就使得解答是不唯一的。Backus 及 Gilbert提出的一种约束条件是，找到一个与初始模型 $\mathscr{W}$ 最靠近的 $\mathscr{W}'$，或 $\mathscr{W}'$ 与 $\mathscr{W}$ 在希尔伯特空间中“距离”最短的作为解答。这种约束条件意味着寻找一个 $\delta_m$ 范数为最小，即 $\|\delta_m\| = \min$ 的解。或者是满足条件

$$\int_{0}^{a} [\delta_m(r)]^2 dr = \min \tag{1-3.5}$$

的解。

若 $\mathscr{W}$ 与 $\mathscr{W}'$ 是最靠近的，则式(1-3.4)式中的 $\varepsilon$ 项可以忽略。用变分法可得

$$\delta_m = \sum_{j=1}^{n} \alpha_j F_j. \tag{1-3.6}$$

式中 $\alpha_j$ 可通过下面 $n$ 个线性方程求得

$$\sum_{k=1}^{n} (F_j,\ F_k)\alpha_k = E_j(\mathscr{W}') - E_j(\mathscr{W}),$$

$$i = 1,\ 2, \cdots,\ n. \tag{1-3.7}$$

式中$(F_j,\ F_k)$是 Férchet 导数的内积。在计算时忽略 $\varepsilon$ 说明这种方法是近似的，即由此得到的$\delta_m$是不精确的。但可以应用迭代法，即每次均以前一次的结果 $\mathscr{W} + \delta_m$ 为起点，逐次得到与实际观测值符合最好的模型。

现在回到反演不唯一性的问题。设真解以 $\mathscr{W}$ 表示，它是所求模型子空间的元素。要表征计算模型反映真实模型的能力可用分辨率衡量。假定在某一半径 $r_0$处，由观测数据得出一个真实模型 $\mathscr{W}_0(r_0)$ 的估计值 $\mathscr{W}(r_0)$，它一般可以看成在 $r_0$ 附近的平均值，即

$$\hat{\mathscr{W}}(r_0) = \int_{0}^{a} A(r, r_0)\mathscr{W}_0(r)dr. \tag{1-3.8}$$

式中$A(r,\ r_0)$称为平均核函数。显然，要使估计值$\hat{\mathscr{W}}(r_0)$最接近真实模型 $\mathscr{W}_0(r_0)$，则应选择$A(r,\ r_0)$为单峰函数，也就是在 $r_0$ 处有一尖锐峰值，而其余 $r$ 处接近于零。理想情况下是选择 Dirac 脉冲函数，即

$$A(r,\ r_0) = \delta(r - r_0). \tag{1-3.9}$$

在这种线性化的约束条件下，平均核函数可以表达为Fréchet 核函数的线性组合，

$$A(r,\ r_0)=\sum_{j=1}^{n} a_j(r_0)F_j(r). \tag{1-3.10}$$

式中，$a_1(r_0)$，$a_2(r_0)$，…，$a_n(r_0)$为与$r_0$点有关的权系数。因此，我们可以自由地选择权系数$a_j(r_0)$来构造最优分辨函数。但是，反演计算得到的模型分辨率(表示唯一性程度)却与误差有相反的关系。有较尖锐峰值的平均核函数$A(r,\ r_0)$尽管反演唯一性较好，却有较大的模型误差。反之，模型误差小则分辨率却较低(平均核函数峰值平缓)。因此，常要在反演的精度与分辨率之间取折衷方案，使二者都得到兼顾。

上面，我们概述了连续型地球模型的 Backus-Gilbert 反演理论。下面，我们将其推广到离散型地球模型中去。

## §1-4　离散型反演问题的数学模型

任何一个地球物理反演问题，诸如根据地球重磁异常计算矿体或构造的空间位置及形态，据根地震体波或面波研究地球内部结构，用地热流或电磁场推算地球内部物质状态，根据空间飞行器轨道计算地球外表形状及内部物质分布，用地磁场资料计算海底扩张及大陆漂移史等，都可以用下述离散模型表述。

由实际观测得到的$m$个离散的地球物理数据，以$m$维向量表示

$$y^0=(y_1^0,\ y_2^0,\ \cdots,\ y_m^0)^T, \tag{1-4.1}$$

它称为观测值向量。

一个具体的地球物理问题，可以用$n$个离散的(具有相同或不同量纲的)物理参数来表示反演所要求的模型，写为$n$维向量，

$$\boldsymbol{x}=(x_1,\ x_2,\ \cdots,\ x_n)^T, \tag{1-4.2}$$

它称为“地球模型”或模型参数向量。

由初始的“地球模型”，根据具体的地球物理问题，计算出一组球物理理论值，它可用$m$维向量表示，

$$y^c(\boldsymbol{x})=(y_1^c(\boldsymbol{x}),\ y_2^c(\boldsymbol{x}),\ \cdots,\ y_m^c(\boldsymbol{x}))^T, \tag{1-4.3}$$

称其为模型的理论值向量。

地球物理反演问题的中心问题是用一定的数学方法，寻找一个“地球模型”，使它尽量与真实地球符合。换言之，即寻找参数向量$\boldsymbol{x}$，使其计算的理论值向量$\boldsymbol{y}^c$与观测值向量$\boldsymbol{y}^0$尽可能符合。为此，可以利用最小二乘原理，构造一个$m$维残差向量，

$$\varepsilon(\boldsymbol{x})=(\varepsilon_1(\boldsymbol{x}),\varepsilon_2(\boldsymbol{x}),\cdots,\varepsilon_m(\boldsymbol{x}))^T. \tag{1-4.4}$$

式中

$$\begin{aligned}
\varepsilon_1(\boldsymbol{x})&=y_1^0-y_1^c(\boldsymbol{x}),\\
\varepsilon_2(\boldsymbol{x})&=y_2^0-y_2^c(\boldsymbol{x}),\\
&\ \ \vdots\qquad\quad \vdots\\
\varepsilon_m(\boldsymbol{x})&=y_m^0-y_m^c(\boldsymbol{x}).
\end{aligned}$$

令残差向量范数平方极小，

$$\|\varepsilon(\boldsymbol{x})\|^2=\sum_{i=1}^{m}\varepsilon_i^2(\boldsymbol{x})=\min. \tag{1-4.5}$$

经过一定的数学变换或采用某些专门的数学方法，可根据(1-4.5)式解出$\boldsymbol{x}$。从而使问题得解。下面举几个地球物理学中常见的例子。

例 1　震源参数的测定

一次地震发生，要求准确测定四个基本震源参数$\lambda$、$\phi$（震中经纬度），$h$（震源深度）及$T$（发震时刻）。这也是一个反演问题。

设有$m(m\geqslant4)$台记录到该次地震，各台观测的到时相应为$t_i^0(i=1,2,\cdots,m)$。

根据震源参数$\lambda$、$\phi$、$h$、$T$的初值（任意假定或粗略估算），可以计算各台的理论到时$t_i^c(\lambda,\ \varphi,\ h,\ T)$，$i=1,2,\cdots,m$。换言之，震源参数（类似于前面的结构参数）决定了理论到时$t_i^c$。求

各台观测到时与理论到时残差，

$$\varepsilon_1 = t_1^0 - t_1^c(\lambda, \phi, h, T),$$
$$\varepsilon_2 = t_2^0 - t_2^c(\lambda, \varphi, h, T),$$
$$\vdots \qquad \vdots$$
$$\varepsilon_m = t_m^0 - t_m^c(\lambda, \phi, h, T).$$

由各台到时残差平方和构造一个目标函数，

$$\Phi(\lambda, \phi, h, T) = \sum_{i=1}^{m} \mu_i \varepsilon_i^2$$

$$= \sum_{i=1}^{m} \mu_i [t_i^0 - t_i^c(\lambda, \phi, h, T)]^2. \qquad (1\text{-}4.7)$$

式中 $\mu_i$ 为权系数，其大小表示评价某一台站数据质量好坏。取目标函数 $\Phi(\lambda,\varphi,h,T) = \min$，由此条件解出四个参数 $\lambda$、$\phi$、$h$、$T$，就达到测定震源参数的目的。

例 2　地震面波及体波反演地壳上地幔结构

地震面波（包括Rayleigh波、Love 波）的相速度或群速度的频散曲线，是地壳上地幔分层速度 $v$、厚度 $h$、密度 $\rho$ 的函数。

设频散曲线或走时曲线有 $m$ 个观测值 $U_1^0$，$U_2^0$，…，$U_m^0$。先假定地壳上地幔分层的结构参数初值 $\boldsymbol{x} = (h_1, v_1, \rho_1; h_2, v_2, \rho_2; \cdots; h_n, v_n, \rho_n)^T$，可以计算出频散曲线或走时曲线上各相应点的理论值 $U_1^c(\boldsymbol{x})$，$U_2^c(\boldsymbol{x})$，…，$U_m^c(\boldsymbol{x})$。将理论计算值与观测值相比较，计算残差

$$\varepsilon_1(\boldsymbol{x}) = U_1^0 - U_1^c(\boldsymbol{x}),$$
$$\varepsilon_2(\boldsymbol{x}) = U_2^0 - U_2^c(\boldsymbol{x}),$$
$$\vdots \qquad \vdots$$
$$\varepsilon_m(\boldsymbol{x}) = U_m^0 - U_m^c(\boldsymbol{x}).$$

同样地，构造一个目标参数，

$$\Phi(\boldsymbol{x}) = \sum_{i=1}^{m} \varepsilon_i^2(\boldsymbol{x}) = \sum_{i=1}^{m} [U_i^0 - U_i^c(\boldsymbol{x})]^2. \qquad (1\text{-}4.8)$$

并令目标函数$\Phi(\boldsymbol{x})=\min$，用一定的数学方法，从中计算出地壳上地幔各分层的参数$\boldsymbol{x}=(h_1, v_1, \rho_1, h_2, v_2, \rho_2, \cdots, h_n, v_n, \rho_n)^T$。

例 3　物探找矿的反演计算问题

重力、地磁、地电的观测异常一般反映了地下介质的不均匀性。如果存在矿体，则可在地面观测到异常值$z_i (i=1, 2, \cdots, m)$。矿体常常可以用模型体来近似表示。这些模型体又通过几何参数和物理参数来描述。例如一个二维有限延伸板状体，其参数就包括板状体中心坐标$x_0$，$z_0$，板的水平宽度$2b$及下延长度$2l$，板的倾角$\alpha$，磁化倾角$i$及磁化强度$J$等七个量。写为$\boldsymbol{x}=(x_0, z_0, 2b, 2l, \alpha, i, J)^T$。

对给定的矿体参数赋以初值，可计算各测点的理论异常值$z_1^c(\boldsymbol{x})$，$z_2^c(\boldsymbol{x})$，$\cdots$，$z_m^c(\boldsymbol{x})$。由残差$\varepsilon_i = z_i^0 - z_i^c(\boldsymbol{x})$，$i=1,2,\cdots,m$，同样可以构造目标函数

$$\Phi(\boldsymbol{x}) = \sum_{i=1}^{m} \varepsilon_i^2 = \sum_{i=1}^{m} [z_i^0 - z_i^c(\boldsymbol{x})]^2. \qquad (1\text{-}4.9)$$

取$\Phi(\boldsymbol{x})=\min$，由此条件同样可以算出待求的矿体参数$\boldsymbol{x}=(x_0, z_0, 2b, 2l, \alpha, i, J)^T$。

## § 1-5　解离散型反演问题的数值方法

目前，用于解离散型地球物理反演问题的数值计算问题经常用到的有广义逆方法以及最优化方法。

上节已经指出，由给定的“地球模型”可以计算一组理论值。若式(1-4.3)中$y^c(\boldsymbol{x})$为线性函数，则其各分量可以写为

$$
\begin{aligned}
y_1^c(\boldsymbol{x}) &= a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n, \\
y_2^c(\boldsymbol{x}) &= a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n, \\
&\vdots \qquad\qquad \vdots \qquad\qquad \vdots \\
y_m^c(\boldsymbol{x}) &= a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n.
\end{aligned}
\tag{1-5.1}
$$

式中$a_{ij}$为常系数。(1-5.1)式也可以直接写为矩阵形式，

$$
\boldsymbol{y}^c(\boldsymbol{x}) = A\boldsymbol{x}. \tag{1-5.2}
$$

其中$\boldsymbol{y}^c$为$m$维向量，$\boldsymbol{x}$为$n$维向量，$A$为$m\times n$矩阵（系数矩阵）。

令残差向量为观测值与理论值之差，

$$
\begin{aligned}
\varepsilon(\boldsymbol{x}) &= \boldsymbol{y}^0 - \boldsymbol{y}^c(\boldsymbol{x}) \\
&= \boldsymbol{y}^0 - A\boldsymbol{x}.
\end{aligned}
$$

于是有关系

$$
A\boldsymbol{x} = \boldsymbol{y}^0 - \varepsilon(\boldsymbol{x}). \tag{1-5.3}
$$

一般说来，地球物理问题所反映的函数关系都是非线性函数。这时，可在某一初始点$\boldsymbol{x}_0$对其进行Taylor展开，并略去二阶及二阶以上高次项，即

$$
\begin{aligned}
y_1^c(\boldsymbol{x}) &= y_1^c(\boldsymbol{x}_0) + \sum_{j=1}^{n}\left(\frac{\partial y_1^c}{\partial x_j}\right)_{\boldsymbol{x}_0}\Delta x_j, \\
y_2^c(\boldsymbol{x}) &= y_2^c(\boldsymbol{x}_0) + \sum_{j=1}^{n}\left(\frac{\partial y_2^c}{\partial x_j}\right)_{\boldsymbol{x}_0}\Delta x_j, \\
&\vdots \qquad\qquad \vdots \\
y_m^c(\boldsymbol{x}) &= y_m^c(\boldsymbol{x}_0) + \sum_{j=1}^{n}\left(\frac{\partial y_m^c}{\partial x_j}\right)_{\boldsymbol{x}_0}\Delta x_j.
\end{aligned}
\tag{1-5.4}
$$

写成矩阵形式为

$$
\boldsymbol{y}^c(\boldsymbol{x}) = \boldsymbol{y}^c(\boldsymbol{x}_0) + A\Delta\boldsymbol{x}. \tag{1-5.5}
$$

式中$A$称为微分系数矩阵(或Jacobi矩阵)，其元素为

$$a_{ij}=\frac{\partial y_i^c}{\partial x_j}\bigg|_{x_0}, \quad \begin{pmatrix} i=1,\ 2,\ \cdots,\ m \\ j=1,\ 2,\ \cdots,\ n \end{pmatrix}.$$

写得更具体是

$$A=\begin{pmatrix} \frac{\partial y_1^c}{\partial x_1} & \frac{\partial y_1^c}{\partial x_2} & \cdots & \frac{\partial y_1^c}{\partial x_n} \\ \frac{\partial y_2^c}{\partial x_1} & \frac{\partial y_2^c}{\partial x_2} & \cdots & \frac{\partial y_2^c}{\partial x_n} \\ \vdots & & & \vdots \\ \frac{\partial y_m^c}{\partial x_1} & \frac{\partial y_m^c}{\partial x_2} & \cdots & \frac{\partial y_m^c}{\partial x_n} \end{pmatrix}_{x=x_0}. \tag{1-5.6}$$

而$\Delta x$为参数$x$的改正量

$$\Delta x=(\Delta x_1,\ \Delta x_2,\ \cdots,\ \Delta x_n)^T. \tag{1-5.7}$$

由(1-5.5)式，可将残差向量写为

$$\varepsilon(x)=y^0-y^c(x)=y^0-y^c(x_0)-A\Delta x. \tag{1-5.8}$$

令向量$b=y^0-y^c(x_0)$表示观测值与初始模型理论计算值之差，则(1-5.8)式可改写为

$$A\Delta x=b-\varepsilon(x). \tag{1-5.9}$$

由上述可见，(1-5.9)式与(1-5.3)式形式上是相似的，即一个非线性问题可以化为线性问题进行计算。所以，今后在计算中只考虑线性方程

$$Ax=b \tag{1-5.10}$$

的解法。对于反演计算为线性函数情况，可直接把$x$作为模型参量，$b$作为观测值。对于非线性函数情况，$x$作为模型参量校正值$\Delta x$，$b$作为观测值与理论计算值之差。在(1-5.10)式中，均略去(1-5.3)及(1-5.9)式中的残差项$\varepsilon(x)$，说明这种计算是近似的。但引入迭代法总可以使这种近似计算逐渐逼近于真解，最后使残差向量范数$\|\varepsilon\|$极小为止。

另一类解地球物理反演问题的数值方法是最优化方法。即直接求目标函数$\Phi(x)$的极小值。其基本思路是用数值方法产生一

个点列

$$\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_k, \cdots,$$

使其满足关系

$$\Phi(\boldsymbol{x}_0) > \Phi(\boldsymbol{x}_1) > \Phi(\boldsymbol{x}_2) > \cdots > \Phi(\boldsymbol{x}_k) > \cdots,$$

直到$\Phi(\boldsymbol{x}_n)$小于预定精度为止。具体的算法如下：

给定一初始点$\boldsymbol{x}_0$，逐次产生寻查方向$\boldsymbol{p}$，并沿$\boldsymbol{p}$方向寻查求目标函数极小点，用迭代格式使$\boldsymbol{x}_i(i=1, 2, \cdots, k, \cdots)$逐渐收敛于极小点。产生寻查方向的方法不同，就构成不同的最优化算法。目前常用的算法大体分为直接寻查法，求一阶导数法及求二阶导数法等三类。具体算法详见第二章。

## §1-6　反演问题的几何模型

对于目标函数$\Phi(\boldsymbol{x})$，可以构造其几何模型，以形象地解释地球物理反演问题的实质。

在$n+1$维超空间中定义一个$n$维超平面，称为"地球模型"参数平面，它上面的一点$\boldsymbol{x}=(x_1, x_2, \cdots, x_n)^T$表示一个具体的地球物理模型。超平面上无穷个点将与无穷个模型相对应。

在这个超空间中，对于一个$\boldsymbol{x}$值，则有一个目标函数$\Phi(\boldsymbol{x})$相对应，所有$\Phi(\boldsymbol{x})$取值点的集合，将构成嵌入$n+1$维超空间的$n$维超曲面，称为目标函数超曲面。在高于三维的抽象空间中，这种几何图形是无法表示的。为此，以二个变量的目标函数空间曲面为例来说明"地球模型"与"目标函数"的关系。图1-6-1中$x_1$及$x_2$为参数平面的两个坐标轴，$x_A$及$x_B$为平面上任意二点。它对应空间曲面上的两点$\Phi_A$及$\Phi_B$。平面上各点与目标函数空间曲面上各点一一对应。

因此，参数超平面上每一点(一个"地球模型")将对应超曲面上一个目标函数。反之，超曲面上一个目标函数值亦与一个"地球模型"(参数平面上的一点)相对应。换言之，在简单的反

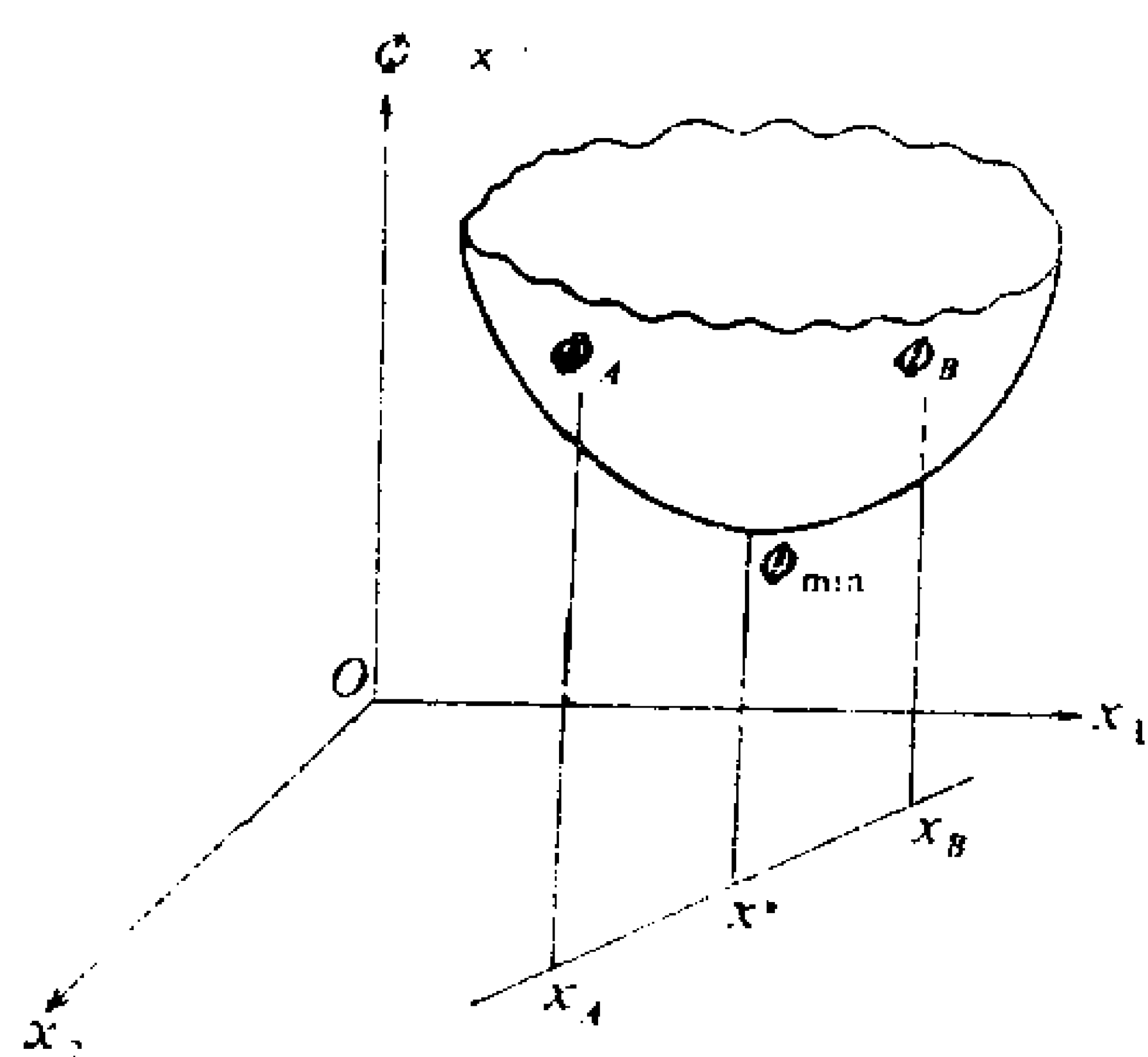



图 1-6-1

演问题中，模型参数 $x$ 与目标函数 $\Phi(x)$ 有一一对应的映照关系。

解地球物理反演问题，就是要寻找超曲面 $\Phi(x)$ 极小点所对应的参数点 $x^*$。但实际上很难找到这个极小点，这首先是一般的目标函数超曲面有着复杂的形状，计算时，常常会找到一些局部的极小点或鞍点。

另一方面，由于任何观测值总存在误差，特别是由于各种干扰因素存在，使实际计算的目标函数 $\Phi(x)$ 也受到畸变。在这种情况下，要找到真正的极小值几乎是不可能的。因此，计算时并不是求真正的极小值，而是求目标函数极小值附近的一个限差范围，设这个限差为 $\varepsilon$，则目标函数 $\Phi(x) \leqslant \varepsilon$ 即满足要求。

显然，这种条件映照在参数平面上的不是一点。而是某一域 $D_\phi$（见图1-6-2）。这意味着凡位于域 $D_\phi$ 中的点，都是反演问题可能解答的参数。即由它们计算的理论值，都能在给定精度要求内拟合观测值。域 $D_\phi$ 愈大，反演问题不唯一性范围愈大。$D_\phi$ 称为反演计算的可行域。

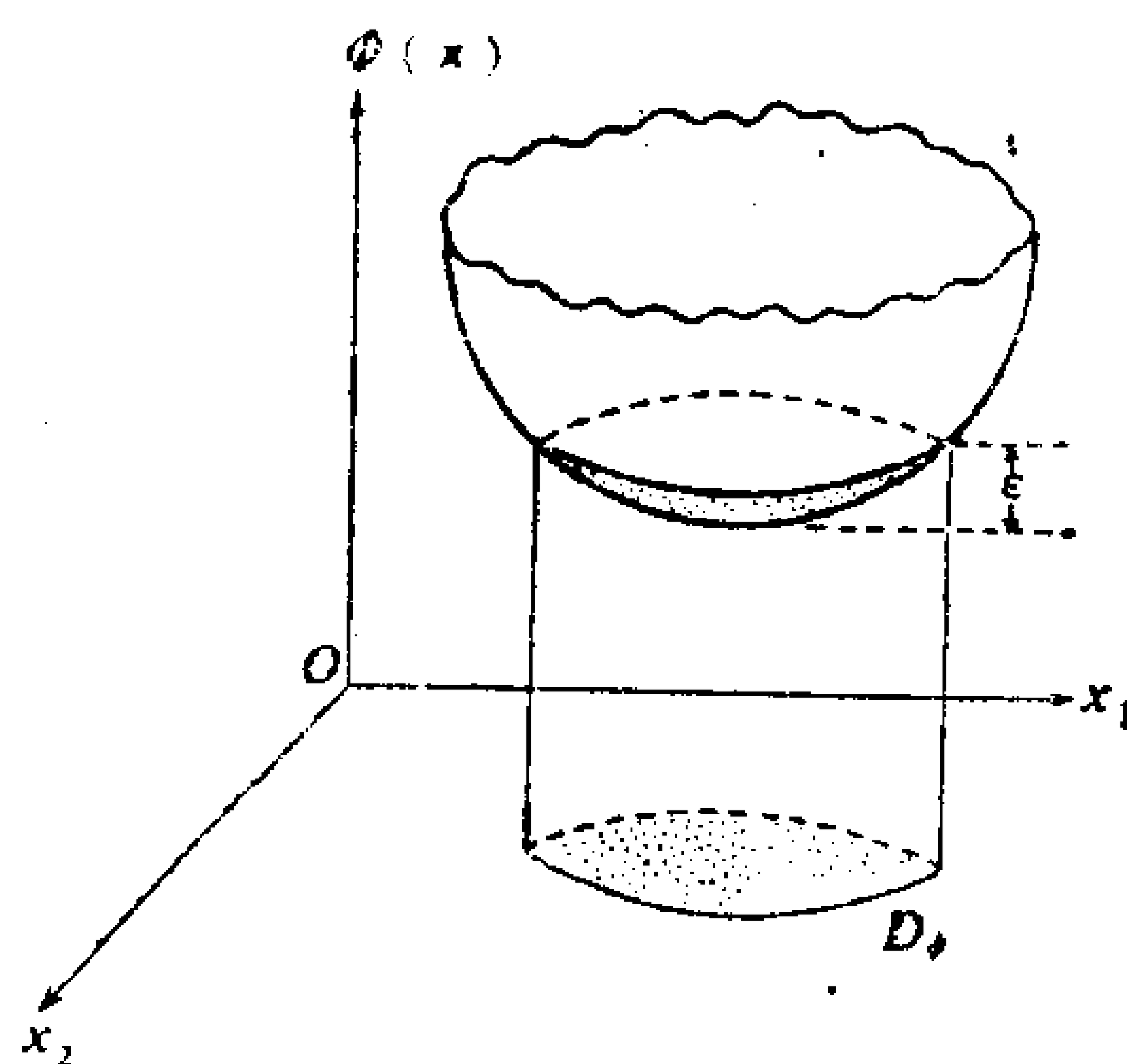



图 1-6-2

## §1-7 反演问题中约束条件的考虑

解地球物理反演问题的不唯一性是由于信息不足及“噪声”存在引起的。要减小这种不唯一性首先就要补充信息，即把经验的(或先验的)及由其它手段提供的资料加入到反演计算中来。有时候，这些补充信息在反演过程中的重要性甚至超过观测数据本身。

“地球模型”先验的或实际的资料构成了反演计算的约束条件，使得参数平面上解答限制在一定范围，称为约束域$D_c$。因此，地球物理反演问题的最终解答实际上是由解的可行域$D_\phi$与约束域$D_c$之交(即$D_\phi \cap D_c$)决定。如图1-7-1所示。显然，要减少解答的不唯一性，缩小域$D_\phi$或$D_c$都能达到目的。

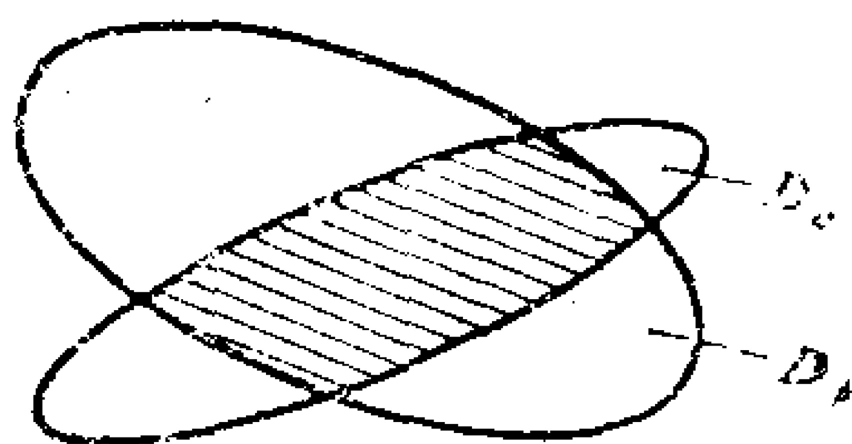


图 1-7-1

由此可见，地球物理反演问题实质上是一个有约束条件的最优化问题，或者是在约束范围内求目标函数极小的问题。事实上，我们总可以由经验的办法或由其它可靠的资料对模型的参数进行约束，以缩小域$D_c$。一般说来，存在着以下五种约束关系。

1．非负约束

地球物质绝大部分物理参数都取正值。例如，地壳及地幔物质的密度，弹性波速度，品质因数，以及地层的厚度，矿体的延伸范围，地球内部的电导率，温度及压力分布等。这种参数的非负性质可表示为

$$x \geqslant 0. \tag{1-7.1}$$

2．区间约束

分布在地球内部一定部位的物质，其物理参数总是在一定范围内取值。例如，在大陆地壳内岩石密度取值范围一般为 1.5≤

$\rho\leqslant3.0$克/厘米$^3$，横波速度取值范围一般为$2.0\leqslant\beta\leqslant4.5$公里/秒。地幔上层的密度取值范围为$3.0\leqslant\rho\leqslant4.5$克/厘米$^3$，横波取值范围为$4.0\leqslant\beta\leqslant6.0$公里/秒。大陆地壳厚度变化范围为$30\leqslant H\leqslant70$公里，海洋区地壳厚度范围一般为$5\leqslant H\leqslant15$公里。其它如介质的品质因数、电导率、磁化率等，均在一定范围内取值。区间约束条件可表示为

$$\boldsymbol{a}\leqslant\boldsymbol{x}\leqslant\boldsymbol{b}. \qquad (1\text{-}7.2)$$

其中参数向量$\boldsymbol{x}=(x_1, x_2, \cdots, x_n)^T$，常数向量$\boldsymbol{a}=(a_1, a_2, \cdots, a_n)^T$及$\boldsymbol{b}=(b_1, b_2, \cdots, b_n)^T$。

3．权平均值约束

地球物质的各种物理参数尽管有千差万别的变化，但在一定区域或一定深度部位，这些参数在总体平均上仍有一定规律。例如，地质构造环境大体相同的区域，地壳的权平均密度

$\sum\limits_{i=1}^{n}\rho_i h_i\Big/\sum\limits_{i=1}^{n}h_i$或权平均速度$\sum\limits_{i=1}^{n}v_i h_i\Big/\sum\limits_{i=1}^{n}h_i$，一般等于某一值或处于某一区间。权平均值约束条件可以写为

$$\boldsymbol{x}^T\boldsymbol{w}=\boldsymbol{b},$$

或

$$\boldsymbol{b}_1\leqslant\boldsymbol{x}^T\boldsymbol{w}\leqslant\boldsymbol{b}_2. \qquad (1\text{-}7.3)$$

式中$\boldsymbol{w}$为权系数向量，$\boldsymbol{b}$，$\boldsymbol{b}_1$及$\boldsymbol{b}_2$为常数向量。

4．函数关系约束

地球物质的各种物理参数不是孤立的，它们之间往往有量的联系(即一定的函数关系)。例如，介质的纵波速度$\alpha$与横波速度$\beta$可由波速比$\gamma$联系起来，

$$\alpha=\gamma\beta. \qquad (1\text{-}7.4)$$

一般地，地壳内$\gamma$的平均值为1.73，上地幔低速层内$\gamma$值可达1.9。由于波速比在一定范围内可看作常数，因而已知其中一个可求出另一个数值。又如，地壳及上地幔物质的密度$\rho$和纵波速度$\alpha$存在一定的统计关系(Talwani，1969；Woollard及其他人，1962，

1969)，二者可用函数形式 $\rho=f(\alpha)$ 表示，其综合关系曲线见图 1-7-2。

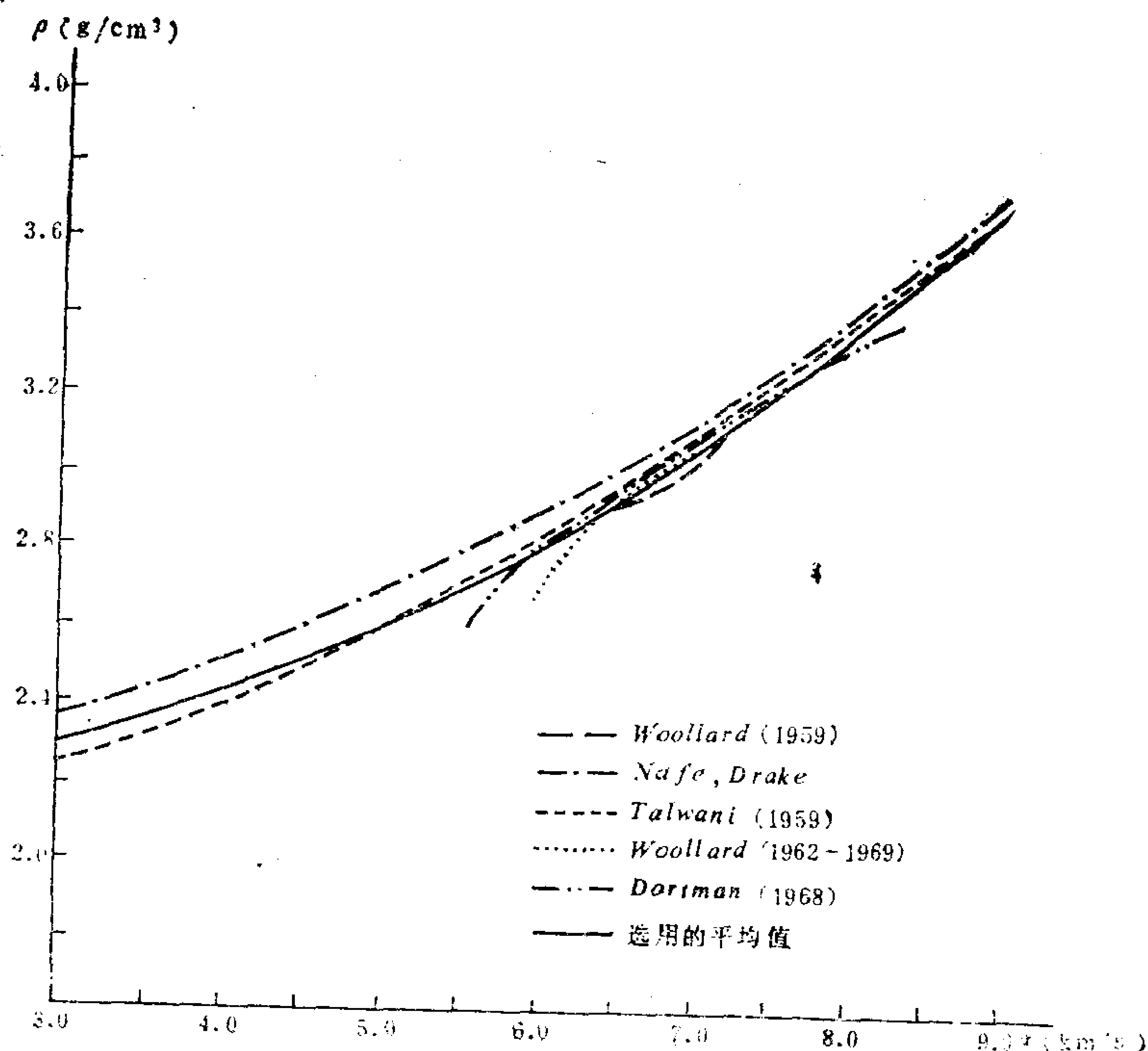



图 1-7-2

这样，利用函数关系约束条件，可由这一组物理参数推算出另一组参数，使反演问题中待求参数减少，简化了求解过程。函数关系约束条件一般写为

$$F(x)=b. \qquad (1\text{-}7.5)$$

5. **特定约束**

上面所讨论的四种约束关系都是先验性质的。但还存在着一类由其它确切的资料（例如由钻井或爆炸地震得到的地层厚度等

参数)得到的约束关系，将地球某些参数作为已知数而加以限定，在作反演计算时，把它作为已知量来处理，从而减少了待求的参数个数。特定约束条件可以写成下列形式，

$$x_i = b_i. \tag{1-7.6}$$

例如，在面波频散曲线的反演中，某些区域的地壳厚度及上地幔顶部的速度可由爆炸地震提供准确资料。这两个参数就作为固定约束量处理。

综上所述，地球物理反演问题的约束条件按其类型可以归为两类，非负约束、区间约束及特定约束可以写为显式约束形式，

$$\boldsymbol{a} \leqslant \boldsymbol{x} \leqslant \boldsymbol{b}; \tag{1-7.7}$$

函数约束及权平均约束可以写为隐式约束形式，

$$c_j \leqslant f_j(\boldsymbol{x}) \leqslant d_j, \quad j = 1, 2, \cdots, p(p \leqslant n). \tag{1-7.8}$$

它可以是线性或非线性函数组。通常我们把由约束条件所限制的集合 $s$ 称为约束可行域，

$$s = \{\boldsymbol{x} \mid a \leqslant x \leqslant b, \ c_j \leqslant f_j(\boldsymbol{x}) \leqslant d_j\} \in R^n. \tag{1-7.9}$$

集合 $s$ 是 $n$ 维空间 $\boldsymbol{R}^n$ 的一个子集。于是约束条件的反演问题就可以表述为：在满足约束式(1-7.7)及(1-7.8)的条件下求目标函数 $\Phi(\boldsymbol{x})$ 极小。或者更一般地抽象为下列形式的数学问题，

$$\left.\begin{array}{l} \min\limits_{\boldsymbol{x} \in s} \ \Phi(\boldsymbol{x}), \\ \\ s = \{\boldsymbol{x} \mid a \leqslant x \leqslant b, \ c_j \leqslant f_j(\boldsymbol{x}) \leqslant d_j\} \end{array}\right\}. \tag{1-7.10}$$

解约束条件下最优化问题称为数学规划。由于目标函数 $\Phi(\boldsymbol{x})$ 及约束函数 $f_i(\boldsymbol{x})$ 中某些函数总是非线性的。所以地球物理反演问题一般可归结为约束非线性规划问题。处理这类问题的方法大体上可以分为两类，一类是直接处理约束条件；另一类是进行某些变换将它化为无约束最优化问题。地球物理反演理论中已愈来愈多地研究这类问题的解法。

# 第二章　解反演问题的基本数值方法

目前，用于解地球物理反演问题的基本数值方法，大都从广义逆理论、最优化方法、及数学规划理论引伸出来。这一章，我们要叙述这些数值方法的原理及算法，并研究如何用来解地球物理反演问题。

## §2-1　广 义 逆 理 论

广义逆理论是线性代数的一个分支，是矩阵论近廿多年的新成就之一。它讨论最一般的矩阵求逆的问题。

### （一）线性方程组的分类

由上一章§1-5的讨论可以看出，用广义逆方法解地球物理反演问题，实际上归结为如何解线性方程组 $A\boldsymbol{x}=\boldsymbol{b}$。其中 $A$ 为 $m\times n$ 矩阵，$\boldsymbol{x}$ 为 $n$ 维列向量，$\boldsymbol{b}$ 为 $m$ 维列向量。根据观测值数目$m$，未知参量的个数$n$，以及$m\times n$矩阵$A$的秩$r=\mathrm{Rank}(A)$这几个量的关系，可以将线性方程组分为以下四类：

**1．自由完全方程组**$(r=m=n)$

这相当于通常的满秩方阵情况。对于任意向量 $\boldsymbol{b}$，都可以得到$x$的精确解。如果 $A$ 为病态的，则可能得到一个不稳定的解答。这类方程组在地球物理反演问题中是不多见的。其结构如图2-1-1中$(a)$所示。

**2．约束完全方程组**$(r=n<m)$

这时观测值个数多于模型参数个数，相当于未知数个数少于

方程个数。这是一个超定方程组的情况。通常，一个给定的超定方程组在数学上是不相容的。对于任意向量 $\boldsymbol{b}$，其解答是不唯一的。仅存在最小二乘意义下的解。由于 $A^TA$ 为对称非奇异，故其逆阵为 $(A^TA)^{-1}A^T$。如果 $(A^TA)$ 是病态的，则导致一个不稳定的解答。这类超定方程组是地球物理反演问题中最大量最常见的。其结构如图2-1-1中(b)所示。

**3. 自由不完全方程组($r=m<n$)**

这时观测值个数少于未知数个数，即相当于方程个数少于未知数的亚定方程组情况。对于任意向量 $\boldsymbol{b}$，解答是不唯一的。这时必须加上其它约束条件，才能求其最小二乘意义下的解。由于 $(AA^T)$ 非奇异，故其逆阵为 $A^T(AA^T)^{-1}$。这类方程组在地球物理反演问题中也经常出现。其结构如图2-1-1中(c)所示。

**4. 约束不完全方程组($r<m, n$)**

这类方程组的秩 $r$ 小于 $m$ 或 $n$，其解答是不唯一的。又分以下几种情况：若 $m=n$ 时，方阵 $A$ 是奇异的(图2-1-1中的(d))，若 $n<m$，别 $A^TA$ 是奇异的(见图2-1-1中的(e))。若 $m<n$，则 $AA^T$ 是奇异的(见图2-1-1中的(f))。这类方程组在地球物理反演问题中是会出现的。

## (二) 广义逆矩阵 $A^-$

我们知道，线性方程组 $A\boldsymbol{x}=\boldsymbol{b}$ 的求解，实质归结为矩阵 $A$ 的求逆问题。这一节，我们要寻求任意 $m\times n$ 矩阵 $A$ 的广义逆矩阵 $A^-$ 的求取方法，并讨论它的性质。我们将按照线性方程组的不同分类，即矩阵 $A$ 的不同的情况分别讨论。

**1. 满秩方阵($r=m=n$)**

即上述的自由完全方程组的系数矩阵 $A$。存在一个唯一的方阵 $A^{-1}$，称为 $A$ 的逆矩阵，满足下列关系

$$A^{-1}A=AA^{-1}=I_n. \qquad (2\text{-}1.1)$$

式中 $I_n$ 为 $n$ 阶单位矩阵，显然有下列性质：

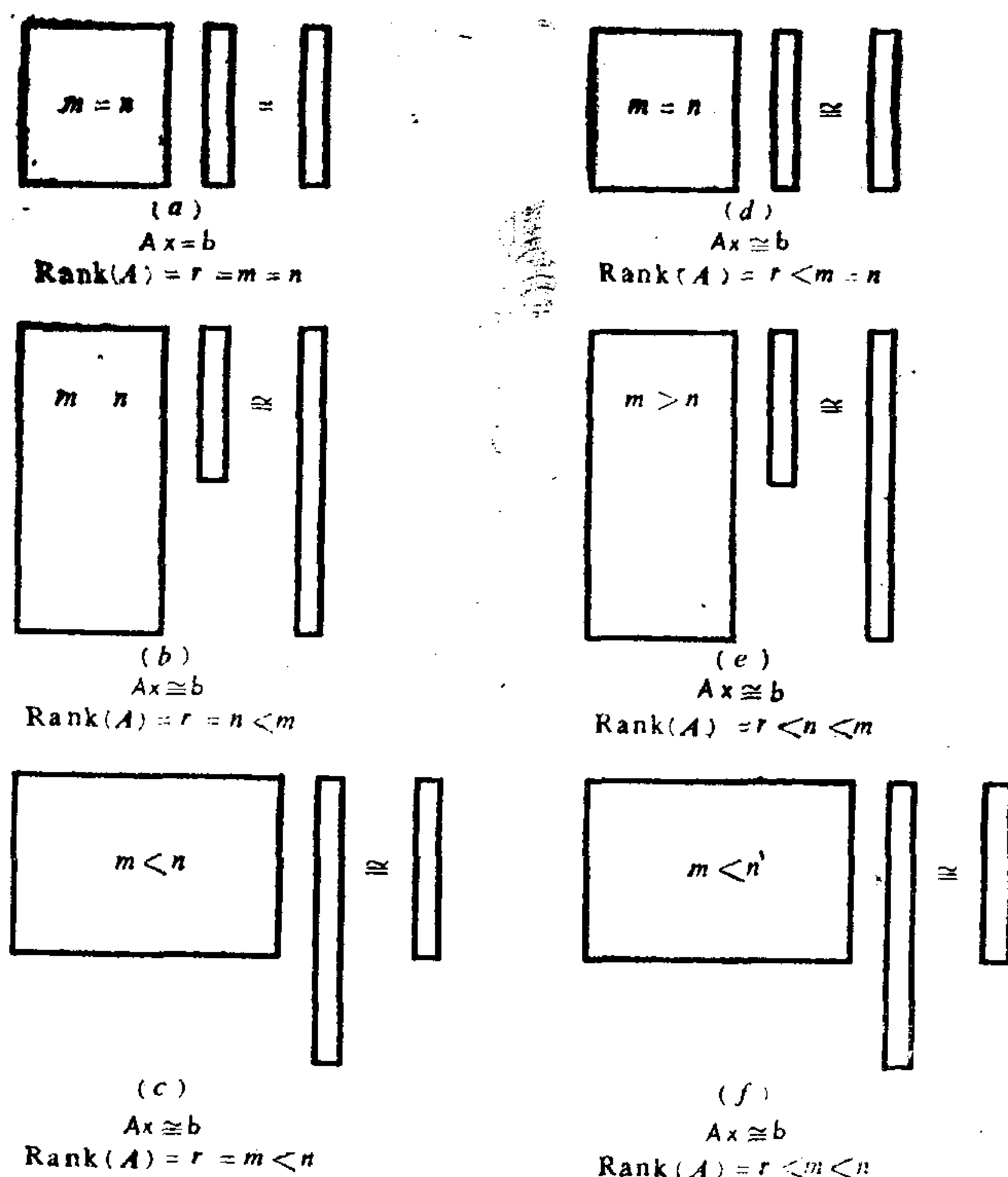



图 2-1-1

$$AA^{-1}A = A,$$
$$A^{-1}AA^{-1} = A^{-1}. \tag{2-1.2}$$

**2. 满秩长方阵**

对于行数$m$与列数$n$不相等的长方阵，只要其秩与行数或列数相等，可以进行所谓满秩长方阵的右逆或左逆的概念。现分两种不同情况讨论：

(1) 行满秩矩阵 ($r = m < n$)

对于秩数与行数相等的行满秩矩阵$A_{m\times n}$，即相当于前述的自

由不完全方程组的系数矩阵具有以下基本性质，

$$\mathrm{Rank}(AA^T)=m. \tag{2-1.3}$$

如果存在一个$n\times m$矩阵$A_R^{-1}$，使$A_R^{-1}$右乘$A$后得到一个$m$阶单位逆矩阵，

$$AA_R^{-1}=I_m, \tag{2-1.4}$$

则称$A_R^{-1}$为$A$的右逆。右逆的计算方法如下，由于

$$(AA^T)(AA^T)^{-1}=I_m, \tag{2-1.5}$$

比较式(2-1.4)及(2-1.5)，可求得右逆

$$A_R^{-1}=A^T(AA^T)^{-1}. \tag{2-1.6}$$

（2） 列满秩矩阵 （$r=n<m$）

对于秩数与列数相等的列满秩矩阵$A_{m\times n}$，即相当于前述的约束完全方程组的系数矩阵，有以下基本性质，

$$\mathrm{Rank}(A^TA)=n. \tag{2-1.7}$$

如果存在一个$n\times m$矩阵$A_L^{-1}$，当$A_L^{-1}$左乘$A$后，得到一个$n$阶单位阵，

$$A_L^{-1}A=I_n, \tag{2-1.8}$$

则称$A_L^{-1}$为$A$的右逆。左逆的计算方法如下，由于

$$(A^TA)^{-1}(A^TA)=I_n. \tag{2-1.9}$$

将式(2-1.9)与(2-1.8)比较，求得左逆

$$A_L^{-1}=(A^TA)^{-1}A^T. \tag{2-1.10}$$

应当指出，当$m\neq n$，左逆$A_L^{-1}$及右逆$A_R^{-1}$是不能同时存在的。仅当$m=n$时，$A_R^{-1}$和$A_L^{-1}$同时存在，并且就等于普通的逆阵$A^{-1}$。

另外，还须指出，满秩长方阵的左逆和右逆是不唯一的。这可由下述定理给出：

对于长方矩阵$A_{m\times n}$，且$\mathrm{Rank}(A)=m$，则矩阵$A$的右逆一般表

达式为

$$A_R^{-1} = VA^T(AVA^T)^{-1}. \qquad (2\text{-}1.11)$$

其中$V$是使得等式 $\text{Rank}(AVA^T) = \text{Rank}(A)$ 成立的任意 $n$ 阶方阵。

同样地，对于长方矩阵$A_{m\times n}$，且$\text{Rank}(A) = n$，则矩阵$A$的左逆一般表达式为

$$A_L^{-1} = (A^TUA)^{-1}A^TU. \qquad (2\text{-}1.12)$$

其中，$U$是使得等式$\text{Rank}(A^TUA) = \text{Rank}(A)$成立的任意$m$阶方阵。

3. 非满秩长方阵 $A$

对于非满秩长方阵 $A_{m\times n}$，即相当于约束不完全方程组的系数矩阵。不再存在式(2-1.3)或式(2-1.7)所给出的基本性质，因此必须引入广义逆$A^-$的概念。

定义　设长方矩阵$A_{m\times n}$的秩$\text{Rank}(A) = r \leqslant \min(m,n)$，则定义矩阵$A$的广义逆$A^-$是满足下面矩阵方程的$n\times m$矩阵

$$AA^-A = A. \qquad (2\text{-}1.13)$$

这是一个最广义的逆矩阵定义，我们把上式定义的广义逆$A^-$称为$g$逆。可以证明，$g$逆$A^-$不但存在，而且一般不唯一。

显然，上述的左逆 $A_L^{-1}$或右逆 $A_R^{-1}$是当$\text{Rank}(A) = r = n$（或$m$）时的$g$逆。而通常意义下的逆阵$A^{-1}$是 $\text{Rank}(A) = r = m = n$ 时的$g$逆。因而广义逆确是普通逆矩阵概念的推广。它把普通逆、左逆、右逆作为一个特殊情形加以概括了。

再来考虑$\text{Rank}(A) = r < \min(m, n)$ 的情况。对于这类非满秩长方阵，存在一个满秩分解定理，即总存在列满秩阵$C_{m\times r}$及行满秩阵$D_{r\times n}$，使得$A_{m\times n} = C_{m\times r}D_{r\times n}$。

于是，显然存在$C$的左逆$C_L^{-1}$及$D$的右逆$D_R^{-1}$。而且广义逆就由下式表示，

$$A^{-}=D_R^{-1}C_L^{-1}. \quad (2\text{-}1.14)$$

在这里，式(2-1. 13)给出了构造$A_{m\times n}$矩阵的 g逆的方法。广义逆$A^{-}$具有以下性质：

(1) $A$的g逆的转置与$A$的转置的g逆是相等的，即

$$(A^{-})^{T}=(A^{T})^{-}. \quad (2\text{-}1.15)$$

(2) 若$(A^TA)^{-}A^T$是$A$的一个g逆，则$A(A^TA)^{-}$是$A^T$的一个g逆。

(3) $A^{-}$的秩不小于$A$的秩

$$\mathrm{Rank}(A^{-})\geqslant \mathrm{Rank}(A). \quad (2\text{-}1.16)$$

(4) $AGA=A$，当且仅当$A^TAGA=A^TA$，其中$G$为$A$的g逆。

## (三) 广义逆$A^{-}$在解线性方程组中的应用

在地球物理反演问题中，所遇到的线性方程组可能是各种类型的。广义逆理论的重要性在于它提供了解各类线性方程组的一般理论及方法。

先来讨论线性方程组的相容性问题。对于一般的方程组

$$\begin{aligned}
&a_{11}x_1+a_{12}x_2+\cdots+a_{1n}x_n=b_1,\\
&a_{21}x_1+a_{22}x_2+\cdots+a_{2n}x_n=b_2,\\
&\cdots \qquad \cdots \qquad \cdots\\
&a_{m1}x_1+a_{m2}x_2+\cdots+a_{mn}x_n=b_m.
\end{aligned} \quad (2\text{-}1.17)$$

或记为

$$A_{m\times n}x_{n\times 1}=b_{m\times 1}. \quad (2\text{-}1.16a)$$

其中，$m$，$n$为任意正整数。若方程式(2-1.17)有解，则称为相容的。否则，称之为不相容或矛盾的。我们在下面讨论中可以看出，广义逆理论把求相容线性方程组的一般解、基本解、最小范数解及求不相容线性方程组的最小二乘解、最小范数最小二乘解的理论全部概括统一起来。

**1. 相容线性方程组的一般解**

定理　对于相容的任意线性方程组 $A\boldsymbol{x}=\boldsymbol{b}$，它的一个特解是

$$\boldsymbol{x}=A^{-}\boldsymbol{b}, \tag{2-1.17}$$

而一般解为

$$\boldsymbol{x}=A^{-}\boldsymbol{b}+(I-A^{-}A)\boldsymbol{c}. \tag{2-1.18}$$

其中 $A^{-}$ 是矩阵 $A$ 的任一个 $g$ 逆，$I$ 为 $n$ 阶单位向量，$\boldsymbol{c}$ 是与 $\boldsymbol{x}$ 同维的任意向量。

这说明，对于一个相容线性方程组 $A\boldsymbol{x}=\boldsymbol{b}$，不论系数矩阵 $A$ 是方阵或长方阵，是满秩的还是降秩的，都有一的个标准的统一求解方法。而且由简洁形式（2-1.18）表达，这是线性方程组理论的重要发展。今后，求任意相容线性方程组，只要求出 $A$ 的一个 $g$ 逆 $A^{-}$ 即可。

**2. 相容线性方程组的最小范数解**

由于 $g$ 逆是不唯一的，因而相容线性方程组的解也不唯一。现在要问，是否存在与 $\boldsymbol{b}$ 无关的某些特殊的 $g$ 逆 $A^{-}$ 使得 $A^{-}\boldsymbol{b}$ 与其它的解相比，具有最小的范数，即

$$\|A^{-}\boldsymbol{b}\|\leqslant\|\boldsymbol{x}\|, \tag{2-1.19}$$

其中 $\boldsymbol{x}$ 是方程组 $A\boldsymbol{x}=\boldsymbol{b}$ 的一切可能解，这一问题可用下述定理表达。

定理　设 $A^{-}$ 为 $A$ 的 $g$ 逆，对任意与 $A$ 相容的 $\boldsymbol{b}$，$A^{-}\boldsymbol{b}$ 在 $A\boldsymbol{x}=\boldsymbol{b}$ 的一切可能解中具有最小范数的充要条件是

$$AA^{-}A=A \text{和} (A^{-}A)^{T}=A^{-}A. \tag{2-1.20}$$

使 $\boldsymbol{x}=A^{-}\boldsymbol{b}$ 具有最小范数的 $A^{-}$ 叫做最小范数 $g$ 逆，常用 $A_m^{-}$ 表示。显然 $A$ 的最小范数 $g$ 逆是用等式 $(A^{-}A)^{T}=A^{-}A$ 对 $g$ 逆加限制得到的。一般说来，最小范数 $g$ 逆不唯一，但是相容方程组的最小范数解却是唯一的。

**3. 不相容方程组的最小二乘解**

由线性方程组理论可知，不相容或矛盾方程组是没有解的，

现在来研究这类方程组的最优近似解问题。

对于不相容的线性方程组$Ax=b$，如果有

$$\|A\hat{x}-b\|\leqslant\|Ax-b\|, \qquad (2\text{-}1.21)$$

则称$\hat{x}$是方程组$Ax=b$在最小二乘意义下的最优近似解，这是因为和任何其它近似解$x$相比较，$\hat{x}$所导致的残差向量范数的平方（或残差平方和）$\|A\hat{x}-b\|^2$是最小的。

定理　设$A^-$是一矩阵，对于任意向量$b$ $A^-b$是方程组$Ax=b$的最小二乘解的充要条件是

$$AA^-A=A\text{和}(AA^-)^T=AA^-. \qquad (2\text{-}1.22)$$

由此可见，$A^-$是一个g逆，称为最小二乘g逆，以$A_l^-$表示。显然，$A$的最小二乘g逆是用等式$(AA^-)^T=AA^-$对g逆加以限制得到的，必须指出，矛盾方程组的最小二乘解导致的误差平方和$\|Ax-b\|^2$是唯一的，但是最小二乘解可以不唯一。

对于地球物理反演问题来说，大多数情况下是解矛盾方程组的问题。因此，只须求出最小二乘的g逆即可，而且，在反演计算中最常见到的一类矩阵是列满秩矩阵，即此矩阵的秩为Rank$(A)=n$。可以证明，列满秩矩阵的左逆

$$A_L^{-1}=(A^TA)^{-1}A^T$$

就是一个最小二乘g逆$A_l^-$。

广义逆理论使得求矛盾方程组最小二乘解的方法标准化、简单化了。整个求解问题归结为求最小二乘g逆$A_l^-$。用不着建立目标函数，求残差平方和，以及根据极值条件求解等一套繁琐步骤。

## （四）Moore–Penrose广义逆$A^+$

上面，对于任意$m\times n$长方阵$A$，由$AA^-A=A$定义了$A$

的 $g$ 逆 $A^-$。这是一个最广义的逆矩阵的概念。我们证明了它的存在，但并不唯一。我们还指出 $A^-$ 在解线性方程组中的作用：$x=A^-b$ 是相容线性方程组 $Ax=b$ 的一个解。然后我们逐步对 $A^-$ 加以限制，得出 $g$ 具有不同性质的子集。

例如，在 $AA^-A=A$ 的基础上，用 $(A^-A)^T=A^-A$ 的条件加以限制后，得出最小范数 $g$ 逆 $A_m^-$，它的性质是 $x=A_m^{-1}b$ 是相容线性方程组 $Ax=b$ 的最小范数解。而在 $AA^-A=A$ 的基础上，用 $(AA^-)^T=AA^-$ 的条件加以限制后，得出最小二乘 $g$ 逆 $A_l^-$，它的重要性质是 $x=A_l^-b$ 为矛盾方程组 $Ax=b$ 的最小二乘解。广义逆 $A^-$ 及其子集 $A_m^-$，$A_l^-$ 等都不是唯一的。下面，我们将对 $g$ 逆作出进一步的限制，定义一个新的广义逆 $A^+$。

定义　设有矩阵 $A_{m\times n}$，对于一个矩阵 $G_{n\times m}$，满足如下四个方程，

$$\begin{aligned} AGA&=A,\\ (GA)^T&=GA,\\ GAG&=G,\\ (AG)^T&=AG. \end{aligned} \tag{2-1.23}$$

则称矩阵 $G$ 为 $A$ 的 Moore-Penrose 广义逆，记作 $A^+$。可以证明，对于任意矩阵 $A$，它的Moore-Penrose广义逆必然存在，且唯一。

广义逆 $A^+$ 的存在证明是构造性的。即在每一种情况下，把满足(2-1.23)式的矩阵 $G$ 直接构造出来。

如果 $A$ 是行满秩矩阵，则其右逆 $A_R^{-1}=A^T(AA^T)^{-1}$ 就是矩阵 $A$ 的Moore-Penrose 广义逆，因为它满足式(2-1.23)的四个关系式。

如果 $A$ 是列满秩矩阵，则其左逆 $A_L^{-1}=(A^TA)^{-1}A^T$ 也是矩阵 $A$ 的Moore-Penrose广义逆，它同样满足式(2-1.23)的四个关系式。

如果 $A$ 是一般的 $m\times n$ 长方矩阵，且 $\mathrm{Rank}(A)=r\leqslant\min(m,$

$n$)。则根据以前的讨论，总可以作出$A$的满秩分解 $A=CD$，其中$C$是$m\times r$ 列满秩矩阵，$D$是 $r\times n$ 行满秩矩阵。利用左逆、右逆定义有

$$
\begin{aligned}
C_L^{-1} &= (C^TC)^{-1}C^T, \\
D_R^{-1} &= D^T(DD^T)^{-1}.
\end{aligned}
\tag{2-1.24}
$$

显然可以验证

$$G = D_R^{-1}C_L^{-1} = D^T(DD^T)^{-1}(C^TC)^{-1}C^T \tag{2-1.25}$$

是$A$的Moore-Penrose广义逆。这样就构造出对任意矩阵$A$均存在的Moore-Penrose广义逆。

关于Moore-Penrose广义逆有如下主要性质：

（1）由广义逆$A^+$的定义知，$A^+$是一个g逆，因此，$x = A^+b$是相容线性方程组 $Ax=b$ 的一个特解，$x = A^+b + I( - A^+A)c$是一般解。$c$是与$x$同维的任意向量。

（2）$(A^+)^T = (A^T)^+$。

（3）$(A^+)^+ = A$.

（4）由$A^+$的定义可知，$A^+$是最小范数 g 逆，因此$x = A^+b$是相容方程组 $Ax=b$ 的最小范数解。

（5）由$A^+$的定义可知，$A^+$是最小二乘 g 逆，因此 $x = A^+b$是矛盾方程组$Ax=b$的最小二乘解。因此，可以引出下面定理。

定理　矛盾方程组 $Ax=b$ 的最小二乘最小范数解为 $x=A^+b$，其中$A^+$是Moore-Penrose广义逆。

由于这一定理，$A^+$又称为$A$的最小二乘最小范数g逆。由于最小二乘最小范数解正是地球物理反演问题所要求的，因此，研究Moore-Penrose广义逆$A^+$的计算就有重要意义。

## §2-2　计算广义逆 $A^+$的正交分解法

在解线性代数方程组时，许多重要的解法，如消元法，直接分

解法及平方根法，改进平方根法等。本质上都是将系数矩阵$\boldsymbol{A}$进行三角形分解。即把一个矩阵化为一个、二个以至三个简单矩阵(三角阵或对角阵)，这样在迭代求解(或矩阵求解)时就很方便。

同样，在计算任意$m\times n$长方矩阵$A$的广义逆$\boldsymbol{A}^+$的各种方法中，目前用得较广的是正交分解法。即将$\boldsymbol{A}_{m\times n}$分解为几个矩阵的乘积，$\boldsymbol{A}^+$就能简单求得。下面给出正交分解定理。

定理　设$\boldsymbol{A}$为任意$m\times n$阶矩阵，其秩$r=\text{Rank}(\boldsymbol{A})$，则总可以把$\boldsymbol{A}$表示成

$$\boldsymbol{A}=\boldsymbol{HRK}^T. \tag{2-2.1}$$

式中　$\boldsymbol{H}$为$m\times m$正交矩阵，

$\boldsymbol{K}$为$n\times n$正交矩阵，

$\boldsymbol{R}$为$m\times n$矩阵，其形式为

$$\boldsymbol{R}=\begin{pmatrix} \boldsymbol{R}_{11} & 0 \\ 0 & 0 \end{pmatrix}_{m\times n}. \tag{2-2.2}$$

$\boldsymbol{R}$矩阵的左上角$\boldsymbol{R}_{11}$为$r$阶非奇异上三角矩阵。

$\boldsymbol{A}_{m\times n}$经正交分解后，则可求出其Moore-Penrose广义逆$A^+$。这可由下述定理表述。

定理　设任意$m\times n$阶矩阵$\boldsymbol{A}$的一种正交分解为

$$\boldsymbol{A}=\boldsymbol{HRK}^T,$$

则其广义逆可以表示为

$$\boldsymbol{A}^+=\boldsymbol{KR}^+\boldsymbol{H}\ . \tag{2-2.3}$$

式中

$$\boldsymbol{R}^+=\begin{pmatrix} \boldsymbol{R}_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix}, \tag{2-2.4}$$

$\boldsymbol{R}_{11}^{-1}$为$r$阶上三角形矩阵$\boldsymbol{R}_{11}$的逆阵。

这一节，主要讨论$QR$分解方法。$QR$分解的基本思想是将任何一个实矩阵通过有限个正交矩阵之积$Q$的作用，把它化为一个简

单矩阵 $\boldsymbol{R}$，这可由以下定理表述。

**定理** 设$\boldsymbol{A}$是一个$m\times n$矩阵，存在一个$m\times m$ 正交矩阵$\boldsymbol{Q}$，使$\boldsymbol{QA}=\boldsymbol{R}$ 成为一个主对角线以下为零的矩阵。

正交变换可以用 Givens 平面旋转变换，也可以用Householder 反射变换。在作Householder 变换时，首先要确定变换矩阵中的向量 $u$。

设给定一个$m$ 维非零向量 $\mathbf{v}$，存在一个正交矩阵$\boldsymbol{Q}$ 使

$$\boldsymbol{Q}\mathbf{v}=-\mathrm{Sign}(v_1)\|\mathbf{v}\|e_1. \tag{2-2.5}$$

其中 $e_1$ 为单位向量基

$$e_1=[1,0,0,\cdots,0]^T,$$

而Sign$(v_1)$表示取向量$\mathbf{v}$第一个分量的符号，即

$$\mathrm{Sign}(v_1)=\begin{cases}+1 & 当v_1\geqslant 0\\ -1 & 当v_1<0.\end{cases}$$

如果定义向量$\boldsymbol{u}$为

$$\boldsymbol{u}=\mathbf{v}+\mathrm{Sign}(v_1)\|\mathbf{v}\|e_1, \tag{2-2.6}$$

可以把Householder反射矩阵写为

$$\boldsymbol{Q}=\boldsymbol{I}_m-\frac{2\boldsymbol{u}\boldsymbol{u}^T}{\boldsymbol{u}^T\boldsymbol{u}}. \tag{2-2.7}$$

由Householder 矩阵镜像变换特性可知，$\boldsymbol{Q}$ 矩阵对向量 $\mathbf{v}$的作用 $\boldsymbol{Q}\mathbf{v}$，相当于将向量 $\mathbf{v}$ 以镜象面 $mn$（$mn$ 垂直于向量 $\mathbf{v}$）映射为 $-\mathrm{Sign}(v_1)\|\mathbf{v}\|e_1$，如图2-2-1示。

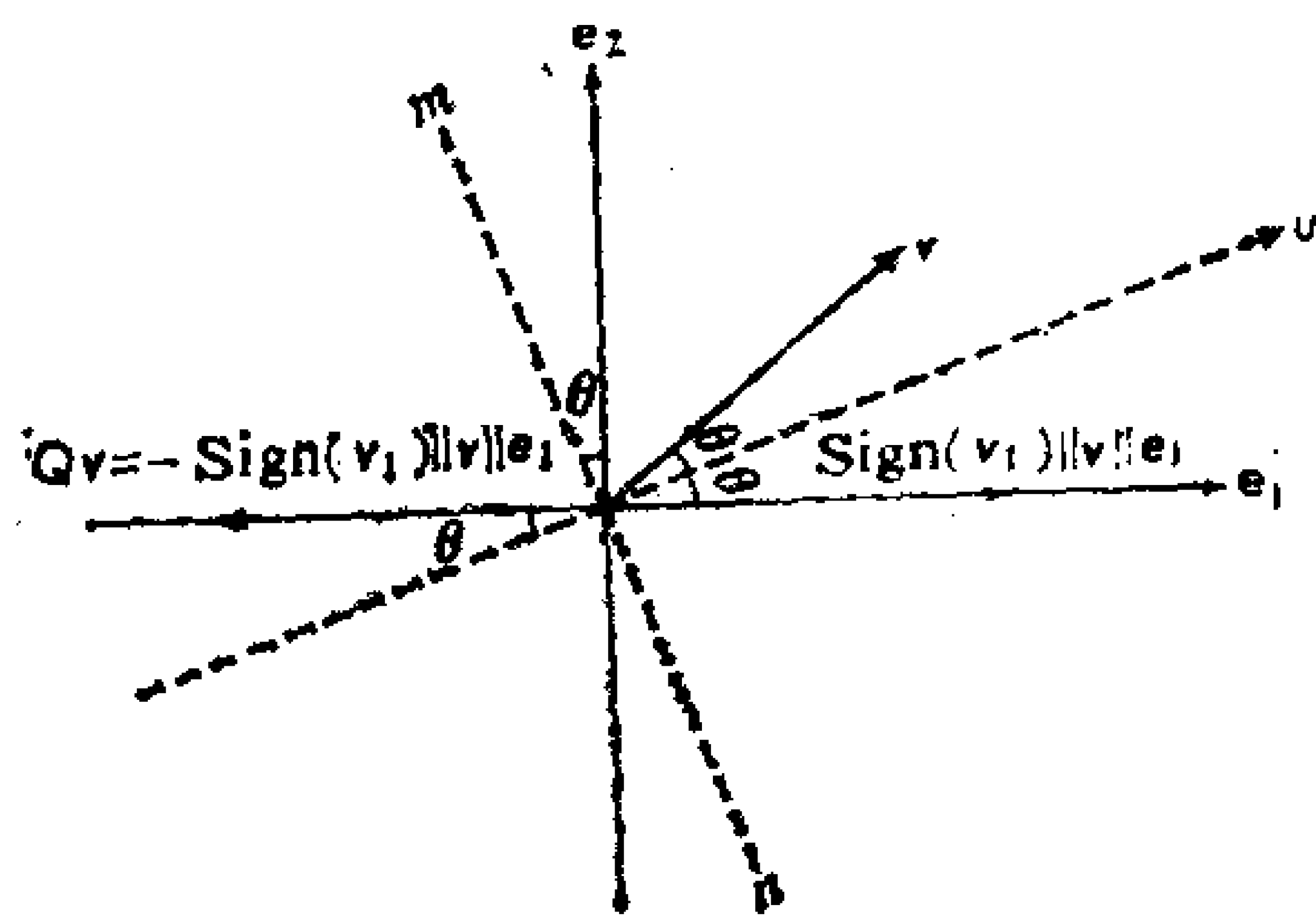



图 2-2-1

这说明，对于任意空间向量$\mathbf{v}$，只要按照式(2-2.6)那样选择向量$\boldsymbol{u}$，构成Householder矩阵，则这种变换可将其映射到单位向量基$e_1$上。换言之，可将$\mathbf{v}$变成为第一个元素等于Sign$(v_1)$ $\|\mathbf{v}\|$，其余元素为零的向量。

于是，可以引出下列构造 $Q$ 矩阵的迭代算法。

第一步，取任意 $m\times n$ 矩阵 $A$ 的第一列的元素作为 $m$ 维向量 $v_1$，以此按(2-2.6)及(2-2.7)式构造 Householder 矩阵 $Q_1$。于是，对于正交变换矩阵 $Q_1A$，第一列除第一个元素外，其余从第2到第$m$个元素都为零。其它各列的元素也将发生变化。

第二步，取$Q_1A$的第二列从第2行到$m$行的元素作为 $m-1$维向量$v_2$，构成Householder矩阵，并令它为 $(m-1)\times(m-1)$阶正交矩阵$P_2$ 构成下列形式正交矩阵

$$Q_2=\begin{bmatrix}1 & 0\\ 0 & P_2\end{bmatrix}. \tag{2-2.8}$$

于是，正交变换矩阵 $Q_2Q_1A$ 的前两列的主对角线以下的元素均变为零。

一般地，对于第$i$步，取 $Q_{i-1}Q_{i-2}\cdots Q_2Q_1A$ 的第$i$ 列从第$i$行到$m$行的元素作为$m-i+1$维向量 $v_i$，以构成 Householder 矩阵,并令它为$(m-i+1)\times(m-i+1)$阶正交矩阵 $P_i$，按下式构成正交矩阵：

$$Q_i=\begin{bmatrix}I_{i-1} & 0\\ 0 & P_i\end{bmatrix}, \tag{2-2.9}$$

式中 $I_{i-1}$ 为$i-1$阶单位矩阵。

显然正交变换矩阵 $Q_iQ_{i-1}\cdots Q_2Q_1A$ 的前 $i$列主对角线以下元素都为零。依此类推作迭代计算，一直到构成正交矩阵$Q_n$，使任意 $m\times n$ 长方矩阵 $A$变为 $Q_nQ_{n-1}\cdots Q_2Q_1A$，即主对角线以下的元素均变为零为止。令

$$Q=Q_{n-1}Q_{n-2}\cdots Q_2Q_1, \tag{2-2.10}$$

注意，$Q_n$ 实际上等于$n$ 阶单位阵 $I_n$，故上式中可略去。

由此可见，用正交矩阵$Q$ 对任意$m\times n$ 矩阵$A$ 进行变换，可得到主对角线以下元素为零的三角阵 $R$

$$QA=R. \tag{2-2.11}$$

这就是线性代数中将任意矩阵$A_{m\times n}$进行分解的 $QR$ 分解法。

下面用这种算法讨论两种矩阵的正交分解。

设$A_{m\times m}$为列满秩矩阵，$n\leqslant m$，Rank($A$)$=n$，据(2-2.11)式可写为

$$A=Q^TR=Q^TRI_n. \tag{2-2.12}$$

式中$Q$为$m\times m$正交矩阵，$R$为$m\times n$上三角阵，$I_n$为$n\times n$单位矩阵。根据矩阵正交分解定理，式(2-2.12)中的$Q^T$、$R$、$I_n$分别与式(2-2.1)中的$H$、$R$、$K^T$矩阵相应。

例如，对列满秩矩阵$A_{m\times n}$($m=3$，$n=2$，Rank($A$)$=2$)，

$$A=\begin{pmatrix} 0.4087 & 0.1594 \\ 0.4302 & 0.3516 \\ 0.6246 & 0.3384 \end{pmatrix}$$

的$QR$分解为

$$Q=\begin{pmatrix} -0.4744 & -0.4993 & -0.7250 \\ 0.5840 & -0.7947 & 0.1652 \\ -0.6587 & -0.3450 & 0.6687 \end{pmatrix};$$

$$R=\begin{pmatrix} -0.8615 & -0.4965 \\ 0.0 & -0.1304 \\ 0.0 & 0.0 \end{pmatrix}.$$

若$A_{m\times n}$为行满秩矩阵，$m\leqslant n$，Rank($A$)$=m$，据(2-2.11)式，可将$A$转置后进行分解，故得

$$A^T=Q^TR, \tag{2-2.13}$$

于是

$$A=R^TQ=I_mR^TQ. \tag{2-2.14}$$

其中$I_m$为$m\times m$单位阵，$R^T$为$m\times n$上三角阵，$Q$为$n\times n$正交矩阵，它们分别相应于正交分解定理中的$H$、$R$、$K^T$矩阵。

例如，对行满秩矩阵$A_{m\times n}$($m=2$，$n=3$，Rank($A$)$=2$)，

$$A=\begin{bmatrix}0.4087 & 0.4301 & 0.6246\\ 0.1594 & 0.3515 & 0.3384\end{bmatrix}$$

的$QR$分解为

$$Q=\begin{pmatrix}-0.4744 & 0.5840 & -0.6587\\ -0.4993 & -0.7947 & -0.3450\\ -0.7250 & 0.1652 & 0.6687\end{pmatrix};$$

$$R=\begin{pmatrix}-0.8615 & 0.0 & 0.0\\ -0.4965 & -0.1304 & 0.0\end{pmatrix}.$$

以上讨论了用 $QR$ 分解对任意 $A_{m\times n}$ 矩阵进行正交分解的办法，有了$A$的正交分解式，容易按(2-2.3)式计算广义逆$A^+$。

## §2-3 计算广义逆$A^+$的奇异值分解法

这一节，我们讨论另一种有实用价值的正交分解法，即任意矩阵$A_{m\times n}$的奇异值分解。这是地球物理反演计算中最常用的一种数值方法。例如见Jackson(1972)，Braile及 Keller(1975)，Aki (1976, 1977), Borsting Pedersen (1977), Vigneresse (1977)等。

上面所讨论的正交分解定理是将$A_{m\times n}$分解为积矩阵$HRK^T$。其中$R$是某种长方阵，其非零元素为非奇异的上三角矩阵$R_{11}$。可以证明，$R_{11}$能进一步简化为非奇异对角矩阵。

这种分解和对称非负定矩阵$A^TA$和$AA^T$的本征值——本征向量分解有密切关系。为了便于理解，我们先讨论$n$阶方阵的正交分解定理。

定理　若$A$为$n$阶非奇异实对称方阵，则总存在正交矩阵$V$，使

$$V^T AV=S, \tag{2-3.1}$$

及

$$A=VSV^T. \tag{2-3.2}$$

式中$S$为对角矩阵，$S=\text{diag}(S_1,S_2,\cdots,S_n)$；$S_1$，$S_2$，$\cdots$，$S_n$

是$A$的本征值。正交矩阵$V$的列向量是方阵$A$的本征向量，且$V$的第$j$列列向量$v_j$为第$j$个本征值$S_j$满足下列关系式，

$$Av_j = S_j v_j. \tag{2-3.3}$$

正交矩阵$V$具有下列性质，

$$V^T V = VV^T = I_n. \tag{2-3.4}$$

根据式(2-3.2)可求出对称方阵$A$的逆矩阵

$$A^{-1} = VS^{-1}V^T, \tag{2-3.5}$$

式中$S^{-1}$是对角元素为$S_i^{-1}$的对角矩阵，即$S=\mathrm{diag}(S_1^{-1}, S_2^{-1}, \cdots, S_n^{-1})$。

如果$A$不是对称方阵，则上述正交分解不成立，但总可以找到两个正交矩阵$U$及$V$，使

$$U^T AV = S, \tag{2-3.6}$$

及

$$A = USV^T. \tag{2-3.7}$$

式中对角矩阵$S$的对角元素$S_1, S_2, \cdots, S_n$是正的非增序列。对角元素的平方$S_1^2, S_2^2, \cdots, S_n^2$是对称矩阵$A^TA$的非零本征值；正交矩阵$V$的列向量$v_j$是对称方阵$A^TA$的本征向量，$v_j$与$S_j^2$满足下列关系，

$$A^TAv_j = S_j^2 v_j. \tag{2-3.8}$$

$U$的列向量$u_j$是对称方阵$AA^T$的本征向量，$u_j$与$S_j^2$满足下列关系：

$$AA^Tu_j = S_j^2 u_j. \tag{2-3.9}$$

应当指出，由(2-3.8)或(2-3.9)式求出的$S_j^2$是相同的。根据式(2-3.7)可以求出任意方阵的逆阵

$$A^{-1} = VS^{-1}U^T. \tag{2-3.10}$$

现在，我们将$n$阶方阵正交分解定理推广到任意$m\times n$矩阵上去。这可由以下奇异值分解(Singular Value Decomposition)定理表述。

定理($SVD$)　设任意$m\times n$矩阵的秩为Rank$(A)=r$，则必存在一个$m\times m$正交矩阵$U$，一个$n\times n$正交矩阵$V$，和一个$m\times n$

对角矩阵$S$，使

$$U^T A V = S, \tag{2-3.11}$$

及

$$A = USV^T. \tag{2-3.12}$$

其中，对角矩阵$S$的主对角线元素恰有 $r$ 个非零，且为正的非增序列。我们称这些对角元素 $S_1$，$S_2$，$\cdots$，$S_r$为$m\times n$ 矩阵 $A$ 的奇异值。这就是奇异值分解名称的来源。

正交矩阵 $U$ 的列向量为 $u_i$，$V$ 的列向量为$v_j$，它们与奇异值 $S_i$ 有以下关系，

$$A^T A v_j = S_j^2 v_j, \quad (j = 1,2,\cdots, n); \tag{2-3.13}$$

$$A A^T u_i = S_i^2 u_i, \quad (i = 1,2,\cdots, m). \tag{2-3.14}$$

且有关系

$$S_i = S_j, \quad 当 i = j \leqslant r;$$

$$S_j = 0, \quad 当 i > r;$$

$$S_j = 0, \quad 当 j > r.$$

下面，我们将奇异值分解式(2-3.12)写得更具体些，

$$A_{m\times n} = U_{m\times m} \begin{pmatrix} S_{n\times n} \\ O_{(m-n)\times n} \end{pmatrix} V^T{}_{\times n}, \tag{2-3.15}$$

其中

$$S_{n\times n} = \begin{pmatrix} S_1 & & & & & & \\ & S_2 & & & 0 & & \\ & & \ddots & & & & \\ & & & S_r & & & \\ & & & & 0 & & \\ & 0 & & & & \ddots & \\ & & & & & & 0 \end{pmatrix}. \tag{2-3.16}$$

对任意矩阵进行奇异值分解，是因为奇异值具有稳定的性质。可以证明，当矩阵$A$有一个扰动$E$ 时，奇异值的变化不会超过$E$的谱范数$\|E\|$，即$E^T E$的最大本征值。奇异值分解的稳定性对用广义逆方法解线性方程组是很有用的。

由奇异值分解式(2-9.12)容易求得任意矩阵$A$的 Moore-

Penrose广义逆为

$$A^{+}=VS^{+}U^{T}. \qquad (2\text{-}3.17)$$

式中$S^{+}$是对角元素为$S_i^{-1}(i=1,2,\cdots, r)$的对角矩阵

$$S^{+}_{n\times m}=\left[\,S^{-1}_{n\times n}\;\vdots\;O_{n\times(m-n)}\right]=\begin{array}{c} \overbrace{\hphantom{S_1^{-1}\quad S_2^{-1}\quad\quad S_r^{-1}\quad 0\quad\quad 0}}^{n}\;\overbrace{\hphantom{000}}^{m-n} \\ \left.\left(\begin{array}{ccccccc:c} S_1^{-1} & & & & & & & \\ & S_2^{-1} & & & 0 & & & \\ & & \ddots & & & & & 0 \\ & & & S_r^{-1} & & & & \\ & 0 & & & 0 & & & \\ & & & & & \ddots & & \\ & & & & & & 0 & \end{array}\right)\right\rbrace n \end{array}.$$

$$(2\text{-}3.18)$$

例如，对于列满秩矩阵($m=3$，$n=2$，Rank ($A$) $=2$)，

$$A=\begin{pmatrix} 0.4087 & 0.1594 \\ 0.4302 & 0.3516 \\ 0.6246 & 0.3384 \end{pmatrix}$$

的奇异值分解为

$$U=\begin{pmatrix} -0.4347 & 0.6141 & -0.6587 \\ -0.5509 & -0.7599 & -0.3450 \\ -0.7125 & 0.2129 & 0.6687 \end{pmatrix},$$

$$S=\begin{pmatrix} 0.9965 & 0.0 \\ 0.0 & 0.1128 \end{pmatrix},$$

$$V=\begin{pmatrix} -0.8626 & 0.5058 \\ -0.5058 & -0.8626 \end{pmatrix}.$$

由于奇异值分解对地球物理反演问题有重要意义，下面将详细讨论它的算法。我们首先介绍用原点平移的 $QR$ 方法对实对称

矩阵进行分解的计算方法。

## （一）加原点平移的 $QR$ 算法

这是一种加快收敛速度的算法。任意 $n$ 阶对称方阵 $\boldsymbol{A}$ 可以通过 $n-2$ 次 Householder 变换或 $(n-1)(n-2)/2$ 次 Givens 旋转变换而变为三对角矩阵。设 $\boldsymbol{A}$ 为 $n\times n$ 实对称方阵，令

$$\boldsymbol{A}_1=\boldsymbol{A}. \tag{2-3.19}$$

取一适当的数 $\sigma_1$（原点平移量），对矩阵 $\boldsymbol{A}_1-\sigma_1\boldsymbol{I}_n$ 作 $QR$ 分解，

$$\boldsymbol{A}_1-\sigma_1\boldsymbol{I}_n=\boldsymbol{Q}_1^T\boldsymbol{R}_1, \tag{2-3.20}$$

然后令

$$\boldsymbol{A}_2=\boldsymbol{R}_1\boldsymbol{Q}_1^T+\sigma_1\boldsymbol{I}_n. \tag{2-3.21}$$

并选取适当的数 $\sigma_2$，对矩阵 $\boldsymbol{A}_2-\sigma_2\boldsymbol{I}_n$ 作 $QR$ 分解，

$$\boldsymbol{A}_2-\sigma_2\boldsymbol{I}_n=\boldsymbol{Q}_2^T\boldsymbol{R}_2, \tag{2-3.22}$$

再令

$$\boldsymbol{A}_3=\boldsymbol{R}_2\boldsymbol{Q}_2^T+\sigma_2\boldsymbol{I}_n, \tag{2-3.23}$$

$$\cdots\cdots.$$

依次迭代计算可得到下列递推公式，

$$\boldsymbol{A}_k-\sigma_k\boldsymbol{I}_n=\boldsymbol{Q}_k^T\boldsymbol{R}_k, \tag{2-3.24}$$

$$\boldsymbol{A}_{k+1}=\boldsymbol{R}_k\boldsymbol{Q}_k^T+\sigma_k\boldsymbol{I}_n, \tag{2-3.25}$$

$$(k=1,2,3,\cdots).$$

我们称 $\sigma_k$ 为第 $k$ 次迭代中的原点平移量，$\boldsymbol{Q}_k$ 是正交矩阵，$\boldsymbol{R}_k$ 是上三角阵，只要 $\boldsymbol{A}_1$ 为三对角阵，则 $\boldsymbol{A}_2$，$\boldsymbol{A}_3\cdots$ 均为三对角阵。实际上，由于对称性，只须计算主对角线及上面的次对角线的元素。

据(2-3.24)式得

$$\boldsymbol{R}_k=\boldsymbol{Q}_k(\boldsymbol{A}_k-\sigma_k\boldsymbol{I}_n), \tag{2-3.26}$$

将其代入(2-3.25)式有

$$\boldsymbol{A}_{k+1}=\boldsymbol{Q}_k(\boldsymbol{A}_k-\sigma_k\boldsymbol{I}_n)\boldsymbol{Q}_k^T+\sigma_k\boldsymbol{I}_n=\boldsymbol{Q}_k\boldsymbol{A}_k\boldsymbol{Q}_k^T. \tag{2-3.27}$$

由（2-3.27）式可知，$A_{k+1}$与$A_k$是相似的，即它们具有相同的本征值$S_1$，$S_2$，…。适当选取平移量$\sigma_k$。可以使$A_k$的最末一行的非对角元素很快地趋于零，而次对角线元素也逐渐趋于零，但较缓慢。这种情况说明，矩阵$A_k$的右下角元素相当迅速地收敛于矩阵$A_1$的一个本征值。

现在讨论$\sigma_k$的取法。如果矩阵$A$为实对称三对角阵，则用QR方法迭代计算的矩阵$A_k$恒保持三对角的特点，这种情况下，平移量$\sigma_k$可以取矩阵$A_k$右下角的2×2的主子矩阵

$$\begin{pmatrix} a_{n-1,n-1}^{(k)} & a_{n-1,n}^{(k)} \\ a_{n,n-1}^{(k)} & a_{n,n}^{(k)} \end{pmatrix} \qquad (2\text{-}3.28)$$

的两个本征值中最接近$a_{n,n}^{(k)}$的那一个。

## （二）奇异值分解的计算

现在我们讨论任意长方矩阵$A_{m\times n}$的奇异值分解(SVD)的计算问题，我们总是假定$m\geqslant n$，若$m<n$、即矩阵$A$的行数少于列数时，可以在$A$阵下面补充$n-m$行零元素，使其成为方阵形式。这些零元素仅需由计算机程序隐含的加以处理，并不要求真正贮存到计算机中去。奇异值分解计算分两步进行。

第一步是用至多$2n-1$次的Householder变换序列将$A$变为上双对角矩阵$\begin{bmatrix} B \\ O \end{bmatrix}$，计算是按下列表达式中用圆括号所规定的秩序进行变换，

$$\begin{bmatrix} B \\ O \end{bmatrix} = Q_n(\cdots Q_2(Q_1 A)H_2)\cdots H_n \equiv Q^T A H. \qquad (2\text{-}3.29)$$

式中

$$Q^T = Q_n Q_{n-1}\cdots Q_1,\quad H = H_2 H_3 \cdots H_n,$$

$$\begin{bmatrix} B \\ 0 \end{bmatrix}_{m\times n} = \begin{bmatrix} B_{n\times n} \\ 0_{(m-n)\times n} \end{bmatrix},$$

及
$$B = \begin{pmatrix} q_1 & e_2 & & \\ & q_2 & \ddots & \\ & & \ddots & e_n \\ & & & q_n \end{pmatrix}_{n\times n}. \qquad (2\text{-}3.30)$$

正交变换阵矩应这样选择：$Q_i$对$A$阵的作用应使它的第 $i$列的$i+1$至$m$行的元素变为零；而$H_i$对$A$阵的作用应使它的第$i-1$行的$i+1$至$n$列的元素变为零。按上节叙述的QR分解方法，总是可以完成这种运算的。

应当指出，$H_n$ 可以简化为单位阵，而仅当 $m=n$ 时 $Q_n$ 为单位阵，$m>n$时 $Q_n$ 一般不能忽略，

SVD计算的第二步是利用加原点平移的 QR 分解法，计算上双对角矩阵$B$的奇异值分解，

$$B = \hat{U} S \hat{V}^T. \qquad (2\text{-}3.31)$$

其中，$\hat{U}$、$\hat{V}$为正交矩阵，$S$为对角阵。由式（2-3.29）可以得出

$$A = Q\begin{bmatrix} B \\ O \end{bmatrix} H^T. \qquad (2\text{-}3.32)$$

将(2-3.31)式代入(2-3．32)式中得到$A$的奇异值分解式为

$$A = (Q\hat{U})\begin{bmatrix} S \\ O \end{bmatrix}(H\hat{V})^T \equiv U\begin{bmatrix} S \\ O \end{bmatrix}V^T. \qquad (2\text{-}3.33)$$

式中
$$U = Q\hat{U},\ V^T = (H\hat{V})^T. \qquad (2\text{-}3.34)$$

现在根据(2-3.30)式来讨论，首先注意到如果某一元素$e_i$为零，则矩阵$B$可以分块作分别讨论。其次我们将指出，如果某一$q_i$为零，则可以应用某种变换以产生分块矩阵。

设$q_k=0$，且$q_j\neq 0$及$e_j\neq 0$，$j=k+1,k+2,\cdots,n$。以$n-k$次Givens 旋转矩阵 $T_j$ 左乘$B$ 得

$$T_n \cdots T_{k+1} B = B' \equiv \begin{bmatrix} q_1 & e_2 & & & & & \\ & \ddots & \ddots & & & & \\ & & q_{k-1} & e_k & & & \\ & & & 0 & 0 & & \\ & & & & q'_{k+1} & e'_{k+2} & \\ & & & & & \ddots & \ddots \\ & & & & & q'_n & e'_n \end{bmatrix}. \quad (2\text{-}3.35)$$

其中$e'_j \neq 0$，$j = k+2$，…，$n$ 及$q'_j \neq 0$，$j = k+1$，…，$n$。

旋转矩阵$T_j$的作用是使$k$行$j$列($j = k+1$，$k+2$，…，$n$)的项为零，且作用于$k$行及$j$列($j = k+1$，$k+2$，…，$n$)。

于是，(2-3.35)式中$B'$具有以下形式

$$B' = \begin{bmatrix} B'_1 & 0 \\ 0 & B'_2 \end{bmatrix}.$$

其中，所有$B'_2$的主对角及次对角元素都不为零。在讨论$B'_2$以前，注意$B'_1$至少有一个等于零的奇异值，这是因为它的右下角元素〔位于$(k, k)$上〕等于零。

接着就是用旋转变换矩阵$R_i$作用于$B'_1$以消去$e_k$，

$$B''_1 = B'_1 R_{k-1} \cdots R_1. \quad (2\text{-}3.35a)$$

式中$R_i$作用于第$i$及第$k$列，使位于$(i, k)$上的元素为零。对于$i>1$，用这种旋转变换使在$(i-1, k)$的位置上的元素不为零。

现在回过来讨论$B'_2$，为了方便仍然用$B$表示这个双对角矩阵，这一矩阵所有的主对角和次对角元素均不为零，符号$n$继续用来表示矩阵$B$的阶。

式(2-3.31)中$B$矩阵的奇异值分解一般采用迭代法计算，令

$$B_1 = B,$$
$$B_2 = U_1^T B_1 V_1,$$
$$\cdots\cdots\cdots.$$

其一般迭代式为

$$B_{k+1} = U_k^T B_k V_k,\ k = 1,\ 2,\ \cdots. \tag{2-3.36}$$

式中，对于所有的$k$，$U_k$及$V_k$均为正交矩阵，$B_k$为上双对角矩阵，选择正交矩阵$U_k$及$V_k$使满足以下关系

$$\tilde{S} = \lim_{k\to\infty} B_k. \tag{2-3.37}$$

式中$\tilde{S}$为对角阵。式(2-3.37)的意义是，经过多次变换后，上双对角矩阵$B$逐渐变为对角阵$\tilde{S}$。

必须指出，由迭代法得出的对角阵$\tilde{S}$的元素一般不一定是正的和按大小次序排列的。为此，要在后面进行适当的整理步骤。

计算奇异值分解的迭代步骤如下。

对式(2-3.37)中每一上双对角阵$B_k$，计算$B_k^T B_k$(这是三对角对称矩阵)右下角$2\times 2$子矩阵的本征值$S_1$和$S_2$，取QR分解中的原点平移量$\sigma_k$等于$S_1$及$S_2$中最接近$B_k^T B_k$三对角阵中右下角元素中的一个。用前述的方法进行分解，就能使三对角阵非主对角元素很快趋于零。

如果正交矩阵$V_k$确定后，则积

$$V_k^T (B_k^T B_k - \sigma_k I_n) \tag{2-3.38}$$

为上三角矩阵。

而正交矩阵$U_k$确定后，则积矩阵

$$B_{k+1} = U_k^T B_k V_k \tag{2-3.39}$$

为上双对角阵。

为了简化记号，我们将去掉式(2-3.38)及(2-3.39)中表示迭

代次数的下标$k$。问题归结为求$U^T$及$V$。

由于$B^TB$是三对角阵，故$V^T(B^TB-\sigma I_n)$为上三角阵，$V^T(B^TB-\sigma I_n)$及$V^T(B^TB)V$均为三对角阵。利用这一性质，可以用Givens旋转矩阵的积来形成(2-3.38)及(2-3.39)中的$V$及$U$，

$$V=R_1R_2\cdots R_{n-1}, \tag{2-3.40}$$

$$U^T=T_{n-1}\cdots T_2T_1. \tag{2-3.41}$$

式中$R_i$作用在矩阵$B$的$i$和$i+1$列，而$T_i$作用在$B$的$i$和$i+1$行。

第一个旋转矩阵$R_1$应选来使$V^T(B^TB-\sigma I_n)$的第一行第一列以下所有的元素都为零。其余的旋转阵$R_i$及$T_i$选来应使$V^T(B^TB)V$为三对角阵。旋转阵依次按$R_1$，$T_1$，$R_2$，$T_2$，$R_3$，$T_3$，…，$R_{n-1}$，$T_{n-1}$的次序确定，并按下列表达式中圆括号的次序作用，

$$T_{n-1}(\cdots T_2((T_1(BR_1))R_2)\cdots R_{n-1}). \tag{2-3.42}$$

旋转阵$T_i(i=1, 2, \cdots, n-1)$作用于第$i$及$i+1$行，并使位置为$(i+1, i)$上的元素为零。旋转阵$R_i(i=1, 2, \cdots, n-1)$作用于第$i$及$i+1$列，并对$i\geqslant 2$时，使位置为$(i-1, i+1)$的元素为零。

这是一个称为追赶法的算法过程。在$n=6$的情况下，上双对角位置以外的元素消去的格式如图2-3-1示。

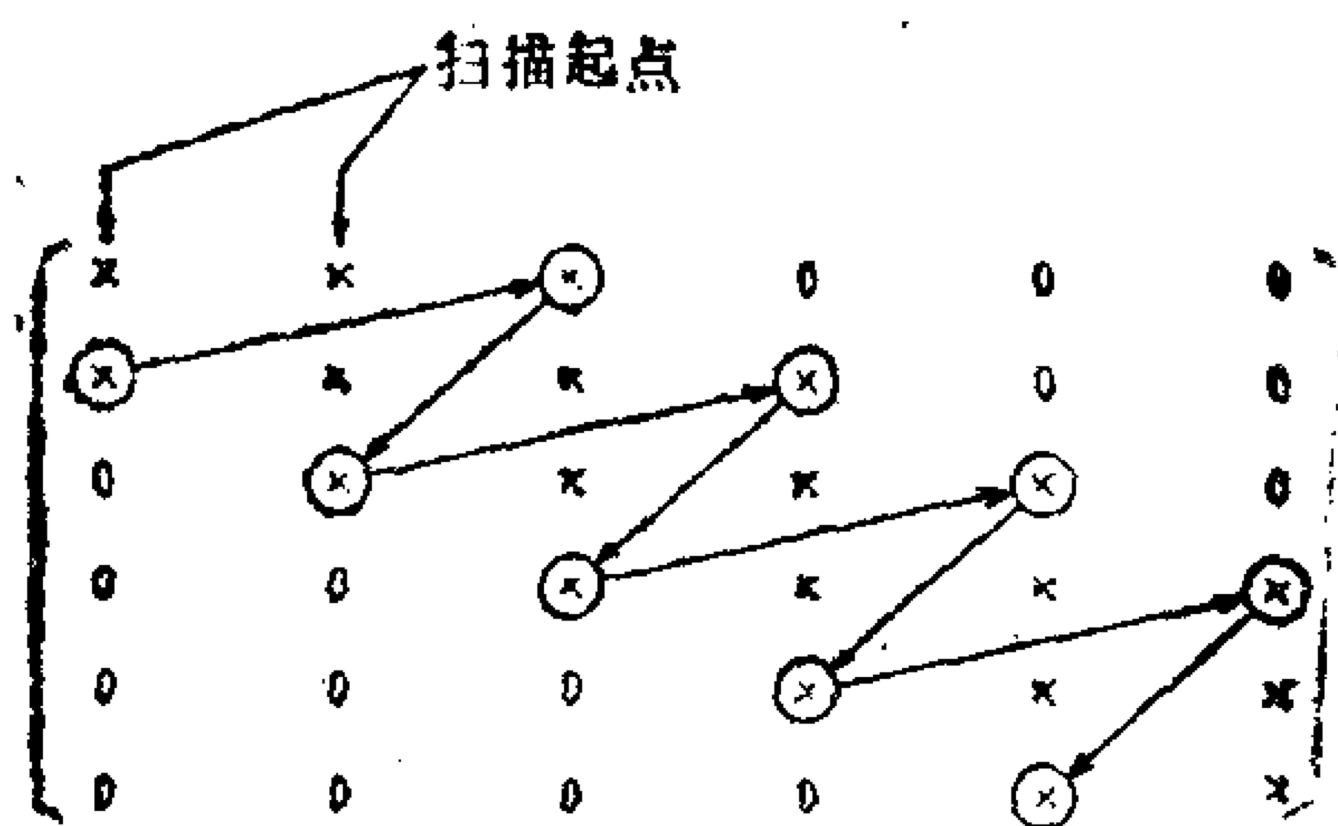



图 2-3-1

由以上讨论可以清楚看到，对于任意$m\times n$矩阵$A$的SVD算法，是基于对(2-3.30)式所定义的非奇异双对角阵$B$的奇异值分解的计算上的。

在计算过程中，要求对式(2-3.30)中次对角线的元素$e_i$进行限差检验。若对任意$i(2\leqslant i\leqslant n)$都有$|e_i|<\varepsilon$($\varepsilon$为给定限差)，则计算结束。否

则将按(2-3.32)式继续进行迭代。最后，以变换后的矩阵$\widetilde{B} = U^T B V$置换原存贮的$B$矩阵而结束。

现在讨论矩阵$A_{m\times n}$奇异值分解的最后公式的一些问题。

由于式(2-3.37)中对角矩阵$\widetilde{S}$的对角元素一般不一定为非负或非增序列。为了补救这一点，可以选择一个对角阵$D$，其对角元素选为+1或-1，使得对角阵

$$\widehat{S} = \widetilde{S} D \tag{2-3.43}$$

的对角元素非负，然后再选择一个排列矩阵$P$（$P$阵中每一列均为单位向量基$e_1$，$e_2$，…，$e_n$按某种方式排列），使矩阵

$$S = P^T \widehat{S} P \tag{2-3.44}$$

的对角元素的排列次序为非增的。容易看出

$$B = (\widetilde{U} P)(P^T \widehat{S} P)(\widetilde{V} DP)^T \equiv \widehat{U} S \widehat{V}^T \tag{2-3.45}$$

是矩阵$B$的奇异值分解，其奇异值矩阵$S$的对角线元素具有非负非增的性质。

于是，对于$A_{m\times n}$的奇异值分解式

$$A = U \begin{bmatrix} S \\ O \end{bmatrix} V$$

中的正交矩阵$V$可按(2-3.34)式计算

$$V = (H\widehat{V}) = H_2 \cdots H_n R_1 \cdots R_\nu DP. \tag{2-3.46}$$

式中$H_i$是式(2-3.29)中的Householder变换，$R_i$是右乘的Givens旋转变换(参见式(2-3.35$a$)，进行$\nu$次以使$e_k \leqslant \varepsilon$为止)。$D$及$P$分别为按式(2-3.43)及(2-3.44)定义的对角阵及排列阵。

而正交矩阵$U$亦按(2-3.34)式计算

$$U^T = (Q\widehat{U})^T = \widehat{U}^T Q^T = PT_\mu \cdots T_1 Q_n \cdots Q_1. \tag{2-3.47}$$

式中$T_i$为按式(2-3.35)定义的左乘Givens旋转阵，$Q_i$为式(2-3.29)中的Householder变换。

最后指出，在式(2-3.46)及(2-3.47)中分别计算$V$及$U^T$，很容易计算$A_{m\times n}$阵的Moore-Penrose广义逆

$$A^+ = VS^{-1}U^T. \tag{2-3.48}$$

## （三）用SVD方法解线性方程组

线性方程组 $Ax \cong b$ 的最小范数最小二乘解为 $x = A^+b$，按(2-3.48)式有

$$x = VS^{-1}U^Tb. \tag{2-3.49}$$

分别令

$$g = U^Tb = \begin{pmatrix} U_1^T \\ U_2^T \end{pmatrix} b \equiv \begin{pmatrix} g^{(1)} \\ g^{(2)} \end{pmatrix} \begin{matrix} \}n \\ \}m-n \end{matrix} \tag{2-3.50}$$

及

$$\hat{p} \cong \begin{bmatrix} S^{-1} \\ 0 \end{bmatrix} g. \tag{2-3.51}$$

因而有

$$x = Vp. \tag{2-3.52}$$

由于 $S$ 为对角矩阵($S = \mathrm{diag}(S_1, S_2, \cdots, S_n)$)，所以式(2-3.51)中的$\hat{p}_i$分量为

$$\hat{p}_i = \frac{g_i}{S_i}.$$

式(2-3.51)可供选择的解向量为

$$P^{(k)} = \begin{pmatrix} \hat{p}_1 \\ \vdots \\ \hat{p}_k \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad k = 0, 1, 2, \cdots, n. \tag{2-3.53}$$

由可供选择的解向量$P^{(k)}$可由(2-3.52)得到$Ax\cong b$可供选择的解答$x^{(k)}$，

$$x^{(k)}=VP^{(k)}=\sum_{j=1}^{k}\widehat{P}_j v^{(j)}\text{，}\ k=0,\ 1,\ \cdots,\ n.$$

(2-3.54)

式中$v^{(j)}$为正交矩阵$V$的第$j$列向量。

注意一般是用以下迭代公式计算$x^{(k)}$

$$x^{(0)}=0,$$

$$x^{(k)}=x^{(k-1)}+P^{(k)}v^{(k)},\ k=1,\ 2,\ \cdots,\ n.$$

(2-3.55)

可供选择解$x^{(k)}$的残差范数平方为

$$\rho_k^2=\|b-Ax^{(k)}\|=\sum_{j=k+1}^{m}g_j^2. \tag{2-3.56}$$

其相应的均方误差为

$$\sigma_k=\sqrt{\frac{\rho_k^2}{m-k}}\text{，}\ k=0,\ 1,\ \cdots,\ n. \tag{2-3.57}$$

## §2-4　最优化方法及一维寻查

上一章已经指出，解地球物理反演问题的数值方法中，一类重要方法是计算目标函数极小值的最优化方法。

最优化方法的研究起源于第二次世界大战期间。当时，由于军事上的需要，研究了一些无法用古典微分法或变分法解决的问题，从而提出了以后称为最优化方法的算法。近卅年来，由于生产及科学研究的需要，大大促进了最优化理论及方法的发展。计算机的普遍使用，使得大型复杂的最优化问题的解决成为可能。

最优化方法这一数学分支包括内容很广，而数学规划是解决最优化问题最重要的数学工具。一般的数学规划问题提法如下：

设$\boldsymbol{x}=(x_1, x_2, \cdots, x_n)^T$为$n$维欧氏的空间$E^n$内的点。而$\Phi(\boldsymbol{x})$；

$$g_i(\boldsymbol{x}), \qquad i=1, 2, \cdots, m;$$

$$h_j(\boldsymbol{x}), \qquad j=1, 2, \cdots p.$$

为给定的$n$元函数。问题是在约束条件$g_i(\boldsymbol{x})\geqslant 0$（不等式的约束）及$h_j(\boldsymbol{x})=0$（等式约束）下求目标函数$\Phi(\boldsymbol{x})$的极小。

若$\Phi$、$g_i$、$h_j$都是$\boldsymbol{x}$的线性函数。我们称为线性规划。若$\Phi$、$g_i$、$h_j$中某些是非线性函数时，称为非线性规划。地球物理反演问题一般是非线性规划问题。

当约束条件不存在，即$m=p=0$时，这类问题称为无约束最优化问题。而当$m$、$p$不全为零时，则称为约束最优化问题。

目前解决约束最优化问题大致有两类方法。其中一类是直接处理约束；另一类是将其化为无约束最优化问题进行处理。由于后一类方法很有效，所以，研究无约束最优化问题就有重要意义。

本节及下一节，将讨论无约束最优化方法的一些基本算法。所谓无约束最优化问题，就是在全空间$E^n$内寻求函数$\Phi(\boldsymbol{x})$的极小点，即求得点$\boldsymbol{x}^*$使

$$\Phi(\boldsymbol{x}^*)=\min\{\Phi(\boldsymbol{x})\}, \qquad \boldsymbol{x}\in E^n.$$

这个问题初看起来很简单，因为若$\Phi(\boldsymbol{x})$有连续的一阶偏导数，令其为零，就可以求得其极小点。但是，解偏导数方程组往往是困难的，一般情况下，是无法直接求解的，因此要用到迭代法。迭代法的基本思路是给定初始点$\boldsymbol{x}_0$并计算一系列的$\boldsymbol{x}_k(k=1, 2, \cdots)$值，使它们收敛于极小点。一般，最优化方法按其求取极小的计算方法可分为直接法和解析法两类。前者仅利用目标函数通过试验来寻查极小点。后者要计算其一阶或二阶导数，通过一定解析式求极小点。下节将分别讨论这些方法。

这里，我们将首先研究无约束最优化计算中一维寻查问题。求目标函数$\Phi(\boldsymbol{x})$的极小点常按以下方式进行：选定初始点$\boldsymbol{x}_0$，从$\boldsymbol{x}_0$沿某一规定的方向$\boldsymbol{p}_0$求函数$\Phi(\boldsymbol{x})$的极小点、设这一点为$\boldsymbol{x}_1$。然后再从$\boldsymbol{x}_1$出发，沿某一规定方向$\boldsymbol{p}_1$求函数$\Phi(\boldsymbol{x})$的极小点，设其为$\boldsymbol{x}_2$。如此继续进行迭代，直到寻查到整体极小点为止。一般说来，从点$\boldsymbol{x}_k$出发，沿规定的方向$\boldsymbol{p}_k$求函数$\Phi(\boldsymbol{x})$的极值点$\boldsymbol{x}_{k+1}$就构成某些最优化方法的一种基本过程。我们称$\boldsymbol{p}_k$为$k$次迭代的寻查方向，它由具体的算法所规定。由图 2-4.1所示，在过$\boldsymbol{x}_k$点的$\boldsymbol{p}_k$方向，函数$\Phi(\boldsymbol{x})$实际上变成了参变量$\lambda$的一元函数$\Phi(\boldsymbol{x}_k+\lambda\boldsymbol{p}_k)$。因为无论$\lambda$取什么值，点$\boldsymbol{x}_k+\lambda\boldsymbol{p}_k$总位于过$\boldsymbol{x}_k$点的$\boldsymbol{p}_k$方向。这种求$\lambda$使$\Phi(\boldsymbol{x}_k+\lambda\boldsymbol{p}_k)$为极小的过程，称为一维寻查或线性寻查。

下面，我们讨论几种常用的一维寻查方法，即研究单个实变量函数$\Phi(\boldsymbol{x})$的极小化问题。

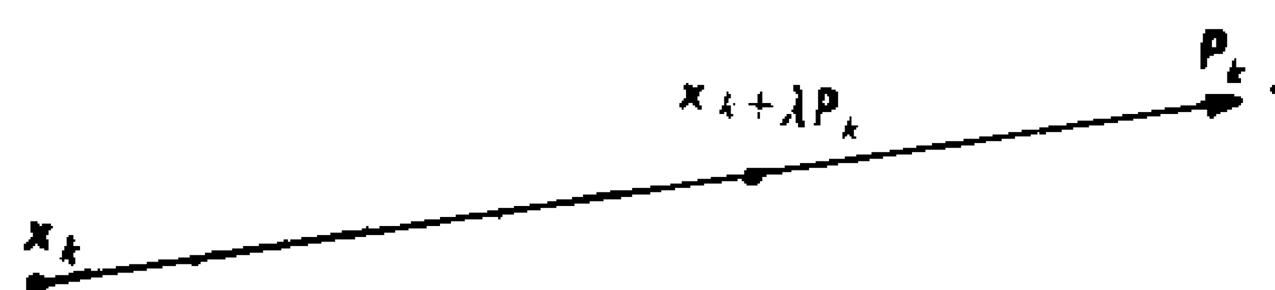



图 2-4-1

## （一）切　线　法

切线法又称 Newton 法，其基本想法是用迭代法产生点列$x_k(k=1, 2, \cdots)$，使得$\Phi'(x_k)$的极限为零。Newton法要求函数$\Phi(x_k)$存在二阶导数。

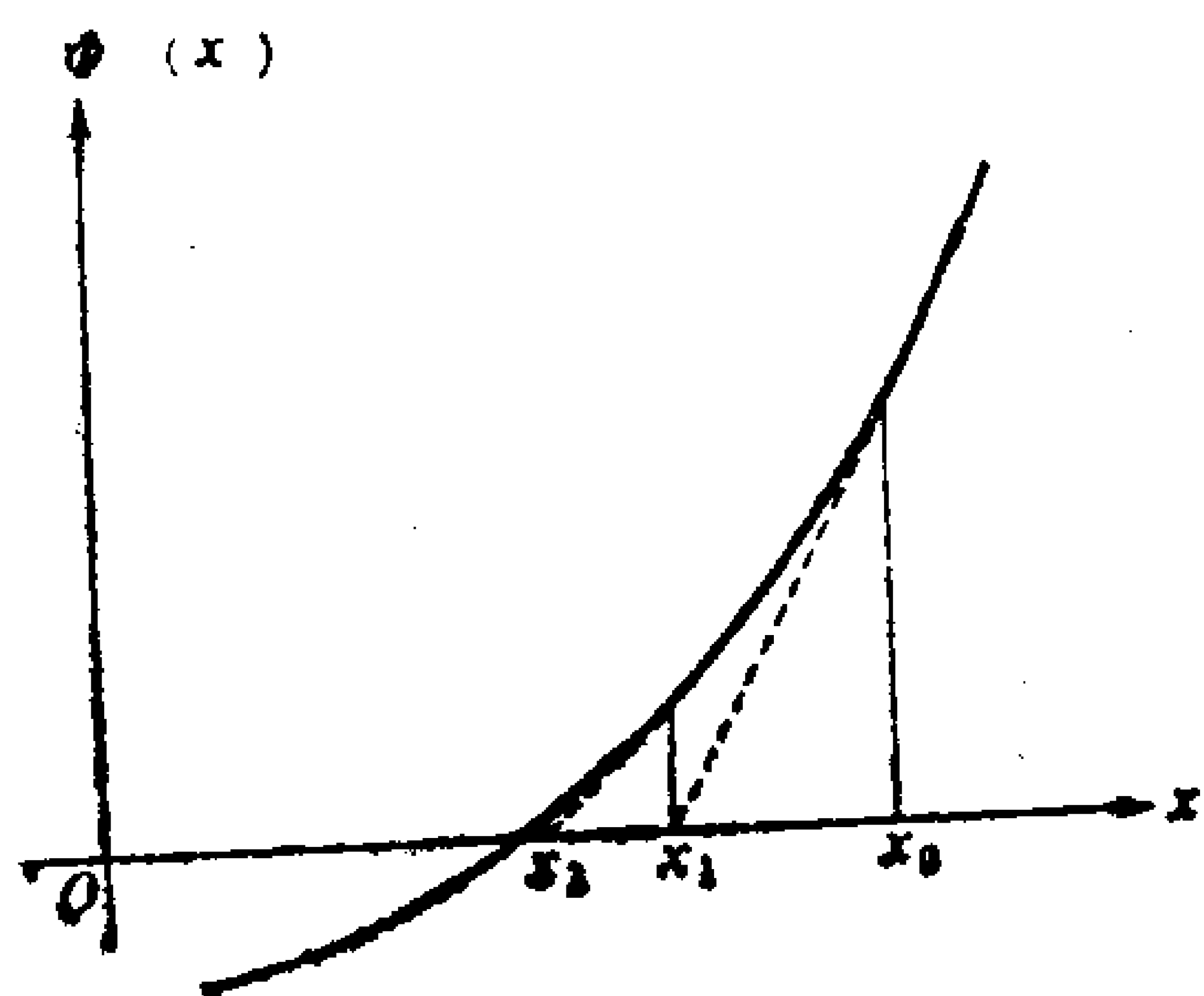



图 2-4-2

如图2-4-2所示，设$x_k$为极小点第$k$次的迭代点。在$x_k$附近使$\Phi'(x)$线性化，并且求得使这个线性函数为零的点。这个点就是下一个迭代点$x_{k+1}$。事实上，若令$x_k$为$\Phi'(x)=0$的解的当前迭代点，线性方程

$$\Phi'(x_k) + \Phi''(x_k)(x - x_k) = 0 \tag{2-4.1}$$

的根为

$$x_{k+1} = x_k - \frac{\Phi'(x_k)}{\Phi''(x_k)}. \tag{2-4.2}$$

当 $|\Phi'(x_k)| \leqslant \varepsilon$ 或 $|x_{k+1} - x_k| \leqslant \delta$，（$\varepsilon$ 及 $\delta$ 均为事先给定的限差）。则取 $x_k$ 作为所求的极小点 $x^*$。若不满足上述的精度要求，则继续迭代，直到满足要求为止。

## （二）多项式逼近法

在一维寻查中，广泛采用多项式逼近方法，即用二次或三次多项式拟合目标函数以求取其极小点。下面只讨论二次函数逼近。

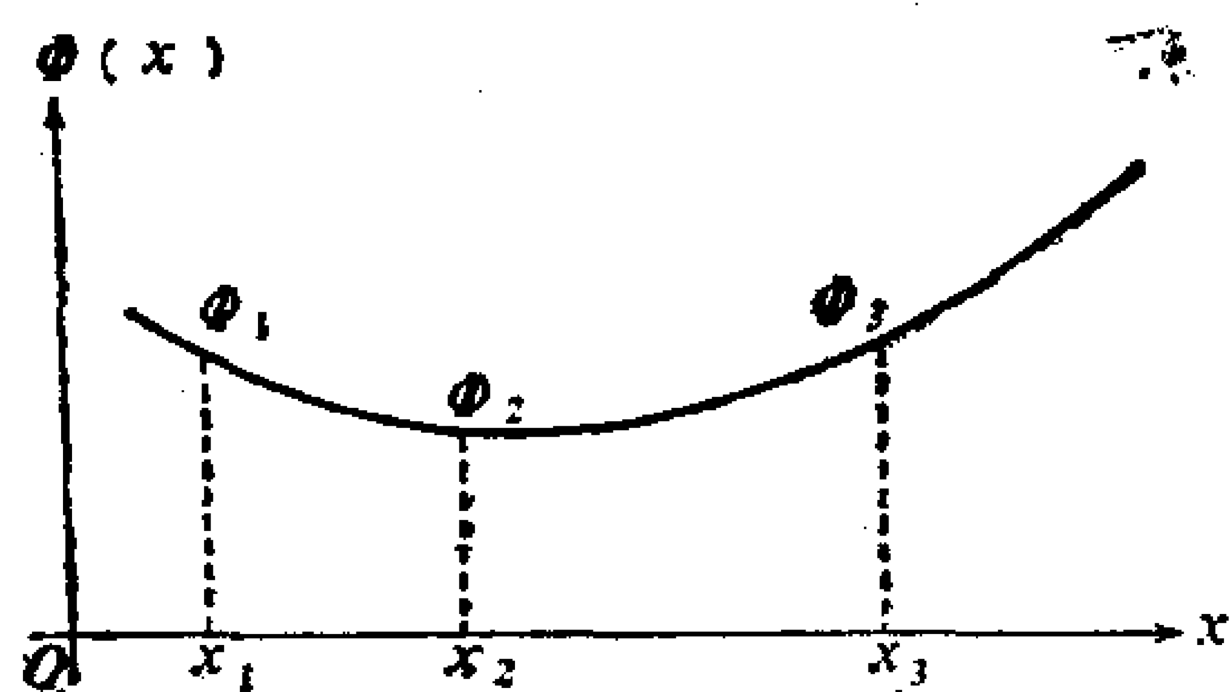



图 2-4-3

如图 2-4-3 所示，设目标函数 $\Phi(x)$ 在三点 $x_1 < x_2 < x_3$ 的函数值分别为 $\Phi_1$，$\Phi_2$，$\Phi_3$。利用这三点的函数值可以构造二次插值公式，求该二次曲线的极小，即可得目标函数 $\Phi$ 的极小值。

令

$$p(x) = a_0 + a_1 x + a_2 x^2 \tag{2-4.3}$$

为二次插值多项式，对通过该二次函数的三点，显然应满足条件，

$$\begin{aligned} p(x_1) &= a_0 + a_1 x_1 + a_2 x_1^2 = \Phi_1, \\ p(x_2) &= a_0 + a_1 x_2 + a_2 x_2^2 = \Phi_2, \\ p(x_3) &= a_0 + a_1 x_3 + a_2 x_3^2 = \Phi_3. \end{aligned} \tag{2-4.4}$$

对于极小点，多项式一阶导数为零，

$$p'(x) = a_1 + 2a_2 x = 0. \tag{2-4.5}$$

故得极小点

$$x_{\min}=-\frac{a_1}{2a_2}. \tag{2-4.6}$$

为解出$a_1$及$a_2$，从(2-4.4)式消去$a_0$得

$$\left.\begin{aligned}a_1(x_1-x_2)+a_2(x_1^2-x_2^2)=\Phi_1-\Phi_2,\\ a_1(x_2-x_3)+a_2(x_2^2-x_3^2)=\Phi_2-\Phi_3.\end{aligned}\right\} \tag{2-4.7}$$

根据克莱姆法则，由(2-4.7)式得

$$a_1=\frac{(x_2^2-x_3^2)\Phi_1+(x_3^2-x_1^2)\Phi_2+(x_1^2-x_2^2)\Phi_3}{(x_1-x_2)(x_2-x_3)(x_3-x_1)},$$

$$a_2=-\frac{(x_2-x_3)\Phi_1+(x_3-x_1)\Phi_2+(x_1-x_2)\Phi_3}{(x_1-x_2)(x_2-x_3)(x_3-x_1)}.$$

将$a_1$及$a_2$代入(2-4.6)式得极小点

$$x_{\min}=\frac{1}{2}\,\frac{(x_2^2-x_3^2)\Phi_1+(x_3^2-x_1^2)\Phi_2+(x_1^2-x_2^2)\Phi_3}{(x_2-x_3)\Phi_1+(x_3-{}_1x)\Phi_2+(x_1-x_2)\Phi_3}. \tag{2-4.8}$$

若$x_1$，$x_2$，$x_3$三点等距，即

$$x_3-x_2=x_2-x_1=\Delta x,$$

则(2-4.8)式可简化为

$$x_{\min}=x_2+\Delta x\frac{\Phi_1-\Phi_3}{2(\Phi_1-2\Phi_2+\Phi_3)}. \tag{2-4.9}$$

应该指出，由这种方法寻查的极小值，仍然是近似的。为了细查，还要利用迭代法。每次迭代时，若满足$a_2>0$(极小点存在条件)，则计算$x_{\min}$。若$x_{\min}$与$x_1$，$x_2$，$x_3$三点中最近的点的距离小于给定精度$\varepsilon$，则取$x_{\min}$作为极小点$x^*$的近似点，否则计算$\Phi(x_{\min})$，并去掉$\Phi_1$、$\Phi_2$及$\Phi_3$中最大的一个，并以余下的二点与$x_{\min}$点重新用二次插值式再寻查极小点，如此继续迭代，直到满足精度要求为止。

## （三）直　接　法

直接法是一种试探法，即利用直接对比函数值$\Phi(x)$的大小来求极小值的方法。例如，可将极小值所在区间$[a, b]$等分为$n$等分，对每一点求目标函数值大小，取最小的一个$\Phi$值对应的$x_{min}$作为$x^*$的近似点。这种方法虽然简单，但计算量很大，所以要考虑如何通过计算最少的函数值以得到$x^*$的近似值。

我们现在来研究单峰函数的性质，它对下面介绍的方法的收敛性和最优性都是非常重要的。

设$\Phi(x)$在区间$[a, b]$上连续，二次可微且$\Phi'(x)=0$只有一个根，那么在$[a, b]$区间内任意取两点$x_k$及$x_k'$，当$a<x_k<x_k'<b$时，有下述关系：

（1）$\Phi(x_k)<\Phi(x_k')$，则极小点在$[a, x_k']$内；

（2）$\Phi(x_k)>\Phi(x_k')$，则极小点在$[x_k, b]$内；

（3）$\Phi(x_k)=\Phi(x_k')$，则极小点在$[x_k, x_k']$内。

对于前两种情况，缩小后的区间里已有一个计算过函数的点，这样我们就可以再找出一个新的点与该点比较，继续缩小区间。为了统一研究方便，我们将第三种情况也并入第二种情况进行研究，以便使缩小后的区间中，也保留一个计算函数的点。

下面介绍的方法称为Fibonacci方法，因为它采用下述的Fibonacci数列来缩小区间。

如果我们把满足精度要求的最小区间视为单位区间。那么要把区间长度为$[a, b]$缩小到单位区间，最少需要计算好多次函数值。或者，反过来这样考虑，我们计算$k$次函数值，最多能把多长的区间$F_k$缩短为单位区间，可以证明$k$与$F_k$有下列关系，

$$F_0=1,\ F_1=1, \qquad (2\text{-}4.10)$$
$$F_k=F_{k-1}+F_{k-2}, \qquad 当k\geqslant 2.$$

具体列表为

| $k$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $F_k$ | 1 | 1 | 2 | 3 | 5 | 8 | 13 | 21 | 34 | 55 | 89 | 144 |

表中 $k$ 表示函数计算次数，$F_k$ 表示最多能缩短为单位区间的长度。我们称$F_k$为Fibonacci 数列。

由上述得知，计算$k$个函数值能获得的最大缩短率为 $1/F_k$，若我们给出的满足精度要求的区间小于 $\delta$，则由

$$\frac{1}{F_k}\leqslant\delta,$$

可以确定最小的计算函数次数$k$。例如，我们要求区间缩短为原来的1/100，那么最少计算函数的次数为11次。

下面讨论用Fibonacci 数列来缩短区间的具体作法。

（1）第一步，根据给定的最小单位区间$\delta$来确定计算函数次数$k$，即按 $\frac{1}{F_k}\leqslant\delta$而确定$F_k$对应的$k$值。

（2）依次按下列迭代方式选择计算函数的点。对于第$i$次探测，由于这时的区间已缩为$[a_i, b_i]$，故取两个新的试点为

$$x_{i+1}=b_i+\frac{F_{k-i-1}}{F_{k-i}}(a_i-b_i),$$

$$x'_{i+1}=a_i+\frac{F_{k-i-1}}{F_{k-i}}(b_i-a_i),$$

$$(i=0,\ 1,\ 2,\ \cdots,\ k-2). \qquad (2\text{-}4.11)$$

判断：

若$\Phi(x_{i+1})<\Phi(x'_{i+1})$，区间缩小为$[a_i, x'_{i+1}]$。

若$\Phi(x_{i+1})\geqslant\Phi(x'_{i+1})$，区间缩小为$[x_{i+1}, b_i]$。

将缩小后的新区间又记为$[a_i, b_i]$，继续迭代，直到$i=k-2$。

（3）当$i=k-2$时，按(2-4.11)式选点得

$$x_{k-1}=b_{k-2}+\frac{1}{2}(a_{k-2}-b_{k-2}),$$

$$x'_{k-1}=a_{k-2}+\frac{1}{2}(b_{k-2}-a_{k-2}).$$

两点完全相同，无法比较。为此，可改为

$$x_{k-1}=b_{k-2}+\left(\frac{1}{2}-\varepsilon\right)(a_{k-2}-b_{k-2}),$$

$$x'_{k-1}=a_{k-2}+\left(\frac{1}{2}+\varepsilon\right)(b_{k-2}-a_{k-2}). \qquad (2\text{-}4.12)$$

式中$\varepsilon$为任意小的正数。与上次留下来的点$x_{k-2}$或$x'_{k-2}$的函数值比较可以得出新的区间，这时得到的新区间即为满足精度$\delta$的单位区间，迭代结束。

下面，我们再介绍一种常用的0.618法(即黄金分割法)。由式(2-4.11)可见，由于Fibonacci法在每次迭代时需要计算比值

$$\frac{F_{k-i-1}}{F_{k-i}},\ (i=0,\ 1,\ 2,\ \cdots,\ k-2).$$

该比值表示每次区间的缩短率，当$i$不同时比值是不同的。由于每次迭代都要算比值，故比较麻烦。另外用这种方法首先要估计$k$的大小，对于某些情况也是困难的。可以证明有极限

$$\lim_{i\to\infty}\frac{F_{i-1}}{F_i}=0.618033988741\cdots. \qquad (2\text{-}4.13)$$

因此就将$\frac{F_{i-1}}{F_i}$，$\frac{F_{i-2}}{F_{i-1}}$，…，$\frac{F_1}{F_2}$比值都用0.618近似的代替，得到下列0.618法的选点格式

$$\begin{aligned}x_{i+1}&=b_i+0.618(a_i-b_i),\\ x'_{i+1}&=a_i+0.618(b_i-a_i).\end{aligned} \qquad (2\text{-}4.14)$$

用与Fibonacci法相同的办法，比较$\Phi(x_{k+1})$及$\Phi(x'_{i+1})$以确定

新区间。不断迭代以缩短区间，直到区间小于$\delta$，迭代过程停止。

0.618法和Fibonacci法探测点的选择都具有对称性，计算过程相似，但Fibonacci法缩小的区间比0.618法要小，当$k$值很大时，0.618法的终止区间要比Fibonacci法长17%，但0.618法计算过程比较简单，而且无须事先计算$k$值。

以上讨论了一些常用的一维寻查方法，下面转入讨论多维无约束最优化方法。我们将按直接计算目标函数值，求一阶导数，求二阶导数三种情况分别叙述。

## §2-5 直接计算目标函数值的方法

直接法由于它无需求目标函数的导数，对于最优化问题计算是很有用的。这类方法内容广泛，我们只介绍最常用的单纯形法及共轭方向法。

### （一）单纯形法

单纯形法是利用对空间参量点的几何图形的研究来寻找目标函数的极小点。单纯形方法出现于六十年代初期，它是在效率很低的网格法的基础上改进的一种经验方法。

一般单纯形取为$n$维空间中$n+1$个互相等距的点组成一个规则几何图形。例如，$n=2$时取等距三点构成等边三角形；$n=3$时取等距四点构成正四面体。对单纯形各顶点计算目标函数$\Phi(x)$的函数值，并将这些值进行比较。舍掉其中函数值最大的点，选择一个新点与原来未舍去的点构成新的单纯形。如此反复作下去，最后总能寻查到极小点。后来Nelder及Mead等对原始的单纯形法进行了改进，其效果较好。这里介绍改进的单纯形法，其步骤如下。

1. 单纯形的选择，可以为等边的规则图形，也可以取不规

则的图形，对于 $n$ 个变量的规则图形的顶点可以由以下公式给出。

设 $\boldsymbol{x}_1^{(0)}$ 是给定的初始点，单位边长的单纯形的顶点为 $\boldsymbol{x}_1^{(0)}$，$\boldsymbol{x}_2^{(0)}$，…，$\boldsymbol{x}_{n+1}^{(0)}$。

其中 $\boldsymbol{x}_1^{(0)}=\left(x_{11}^{(0)},\ x_{12}^{(0)},\ \cdots,\ x_{1n}^{(0)}\right)^T$，

$$\boldsymbol{x}_i^{(0)}=\left(x_{11}^{(0)}+\delta_1,\ x_{12}^{(0)}+\delta_1,\ \cdots,\ x_{1\,i-1}^{(0)}+\delta_1,\right.$$

$$\left.x_{1i}^{(0)}+\delta_2,\ x_{1\,i+1}^{(0)}+\delta_1,\ \cdots,\ x_{1n}^{(0)}+\delta_1\right)^T,$$

$$i=2,\ 3,\ \cdots,\ n+1. \qquad (2\text{-}5.1)$$

及

$$\delta_1=\frac{\sqrt{n+1}+n-1}{n\sqrt{2}},$$

$$\delta_2=\frac{\sqrt{n+1}-1}{n\sqrt{2}}. \qquad (2\text{-}5.2)$$

对于不等边单纯形构成法：除初始点 $\boldsymbol{x}_1^{(0)}$ 外，可以从该点起始沿每一个坐标轴方向各取一点，即 $\boldsymbol{x}_{i+1}^{(0)}=\boldsymbol{x}_1^{(0)}+\lambda_i\boldsymbol{e}_i(i=1,\ 2,\ \cdots,\ n)$。共$n+1$个点构成初始单纯形。其中 $\lambda_i$ 为适当的标量。

2．计算单纯形各顶点的函数值 $\Phi(\boldsymbol{x}_i)$，并按目标函数值大小重排顺序，

$$\begin{array}{l}\Phi_1,\ \Phi_2,\ \cdots,\ \Phi_n,\ \Phi_{n+1}.\\ \boldsymbol{x}_1,\ \boldsymbol{x}_2,\ \cdots,\ \boldsymbol{x}_n,\ \boldsymbol{x}_{n+1}.\end{array} \qquad (2\text{-}5.3)$$

其中 $\Phi_1<\Phi_2<\cdots<\Phi_n<\Phi_{n+1}$。一般称目标函数最小点 $\boldsymbol{x}_1$ 为最好顶点，而目标函数最大点 $\boldsymbol{x}_{n+1}$ 为最坏顶点，$\boldsymbol{x}_n$ 为次坏顶点。

不考虑最坏顶点，计算单纯形顶点$\boldsymbol{x}_1$，$\boldsymbol{x}_2$，…，$\boldsymbol{x}_n$的质量中心，以$c=(c_1, c_2, \cdots, c_n)^T$表示，其各分量是计算公式为

$$c_i=\frac{1}{n}\sum_{j=1}^{n}x_{ji},\ (i=1,\ 2,\ \cdots,\ n). \tag{2-5.4}$$

3．将$\boldsymbol{x}_{n+1}$顶点去掉，代之以它的反射点$\boldsymbol{x}_r$，反射点的计算公式为

$$\boldsymbol{x}_r=\boldsymbol{c}+\alpha(\boldsymbol{c}-\boldsymbol{x}_{n+1}). \tag{2-5.5}$$

式中 $\alpha$ 称为反射系数。

对反射点 $\boldsymbol{x}_r$ 处的目标函数值的计算有三种可能情况。

(1) $\Phi(\boldsymbol{x}_1)<\Phi(\boldsymbol{x}_r)<\Phi(\boldsymbol{x}_n)$，这时以$\boldsymbol{x}_r$代替$\boldsymbol{x}_{n+1}$。按正常情况继续迭代。

(2) $\Phi(\boldsymbol{x}_r)<\Phi(\boldsymbol{x}_1)$，这时反射点$\boldsymbol{x}_r$为新的最好点。可以将 $\boldsymbol{x}_r$ 沿此方向再向前推进一些，称为扩张，即取扩张点

$$\boldsymbol{x}_e=\boldsymbol{c}+\beta(\boldsymbol{x}_r-\boldsymbol{c}). \tag{2-5.6}$$

式中 $\beta>1$称为扩张系数。计算$\Phi(\boldsymbol{x}_e)$，若$\Phi(\boldsymbol{x}_e)<\Phi(\boldsymbol{x}_r)$，则称为扩张成功，以$\boldsymbol{x}_e$代替$\boldsymbol{x}_{n+1}$。否则扩张失败，仍以 $\boldsymbol{x}_r$ 代替$\boldsymbol{x}_{n+1}$。继续进行迭代。

(3) $\Phi(\boldsymbol{x}_n)\Phi<(\boldsymbol{x}_r)$，即 $\boldsymbol{x}_r$ 为新的最坏点时，可以进行压缩。取压缩点

$$\boldsymbol{x}_g=\boldsymbol{c}+\gamma(\boldsymbol{c}-\boldsymbol{x}_{n+1}). \tag{2-5.7}$$

式中 $\gamma\ (0<\gamma<1)$称为压缩系数。如果计算出的目标函数值满足

$$\Phi(\boldsymbol{x}_g)<\min(\Phi(\boldsymbol{x}_r),\ \Phi(\boldsymbol{x}_{n+1})), \tag{2-5.8}$$

则称压缩成功。以压缩点$\boldsymbol{x}_g$代替最坏点$\boldsymbol{x}_{n+1}$。否则，将各顶点与最好点的距离折半，缩小单纯形，向最好点靠近。重新计算各顶点的函数值，继续进行迭代。

图2-5-1为两个参量的单纯形法计算过程中，反射、扩张与压缩的示意图。

4．收敛准则。当单纯形各顶点的函数值的标准偏差 $\sigma$ 小于事先给定的精度要求 $\varepsilon$ 时，迭代过程收敛，停止计算。标准偏差

$\sigma$ 的计算公式为

$$\sigma=\sqrt{\frac{\sum_{i=1}^{n+1}(\Phi(x_i)-\overline{\Phi(x)})^2}{n}}. \tag{2-5.9}$$

式中$\overline{\Phi(x)}$为单纯形各顶点目标函数值的平均值。

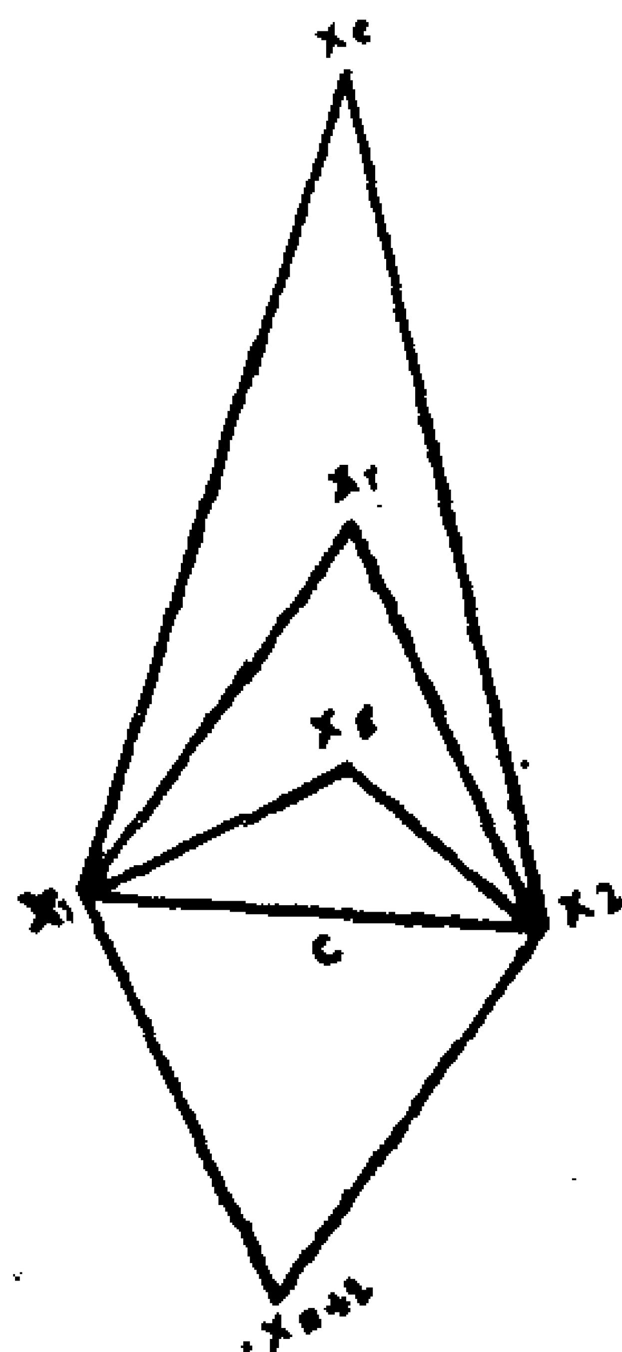


图 2-5-1

单纯形法不仅对无约束最优化问题有效。对于约束最优化也是一种很好的算法。这时仅需对每次迭代中破坏了约束条件的顶点，赋以一个较大的函数值（比所有顶点都大），这就可以使迭代计算过程逐渐远离边界。

单纯形法计算简单，程序稳定。但参量较多时，计算工作量大。

## （二）共轭方向法

下面讨论目标函数为二次函数时共轭方向的重要特性，它是整个一类无约束最优化求极小值的基础。

定义　设$\boldsymbol{A}$为$n\times n$正定对称矩阵，$\boldsymbol{p}_1$，$\boldsymbol{p}_2$ 为两个$n$维向量，若有

$$\boldsymbol{p}_1^T\boldsymbol{A}\boldsymbol{p}_2=0, \tag{2-5.10}$$

则称$\boldsymbol{p}_1$和$\boldsymbol{p}_2$为$\boldsymbol{A}$共轭或关于$\boldsymbol{A}$的共轭方向。

共轭方向的概念是正交性（$\boldsymbol{A}$为$n\times n$单位矩阵$\boldsymbol{I}_n$）的一个推广。一个对称的$n\times n$方阵$\boldsymbol{A}$具有$n$个正交的本征向量，这几个向量也是相互共轭的。因此，对每个$n\times n$正定对称方阵，至少有$n$个相互共轭的方向。可以证明，对于一个二次型函数，沿这

$n$个共轭方向求极小，至多$n$次就可以收敛到极小点 $x^*$。因此，利用共轭方向来寻找极小是方便的。下面介绍一般的共轭方向法。

首先，给定一个初始点 $x_0$，并任意建立初始方向。通常，初始寻查方向均选为$n$ 维空间的坐标基(这是一组线性无关方向)，即

$$
\begin{aligned}
p_1 &= e_1 = (1,0,0,\cdots,0)^T, \\
p_2 &= e_2 = (0,1,0,\cdots,0)^T, \\
&\vdots \qquad\qquad \vdots \\
p_n &= e_n = (0,0,0,\cdots,1)^T.
\end{aligned}
$$

对上述每一个方向依次进行一维寻查，即求$\lambda_i$，使

$$\Phi(x_{i-1}+\lambda_i p_i)=\min_{\lambda}\Phi(x_{i-1}+\lambda p_i).$$

并令

$$x_i = x_{i-1}+\lambda_i p_i.$$

重复此寻查过程，直到$i=n$，即对所有的方向 $e_i$，都寻查完为止。这时得到点$x_n$。连接$x_0$与$x_n$方向，得到一个共轭方向

$$p_n' = x_n - x_0. \qquad (2\text{-}5.11)$$

将原方向 $p_1$ 取消，并将$p_n'$补在后面。换言之，也就是令$p_2$，$p_3$，$\cdots$，$p_n$，$p_n'$作为 $n$ 个寻查方向，以$x_n$ 作为起点再依次作一维寻查。最后得到另一个新点$x_n'$和另一个共轭方向$p_n''$。

继续按此步骤计算。每迭代一次，就增加一个共轭方向。迭代$n$ 后次，就可以得到$n$ 个共轭方向。对于二次型目标函数依次对这$n$ 个共轭方向进行一维寻查，即可求得极小点$x^*$。

## （三）Powell法

在以上方法中，存在一个缺点。就是$n$ 个方向中的任意一个方向上，若 $\lambda_i=0$，这时便不能保证$n$个方向是线性无关的，为此，Powell 提出了一种改进方法。这种共轭方向法的基本思

想是，在初始方向$\boldsymbol{p}_i(i=1,2,\cdots,n)$上，依次进行单方向探测求极小点，将初始点与第$n$个方向上的极小点的连线方向作为新的方向。如果该方向代替$\boldsymbol{p}_i(i=1,2,\cdots,n)$中的某一方向，能使由各$\boldsymbol{p}_i$分量作为第$i$列元素构成的行列式增大，则去掉该方向，代之以新方向。否则，仍不能代替。其具体算法如下：

1. 选定初始值$\boldsymbol{x}_0$及初始寻查方向。一般是选取$n$个有顺序的线性独立方向$\boldsymbol{p}_1$，$\boldsymbol{p}_2$，…，$\boldsymbol{p}_n$，即取$n$维空间的坐标基$e_1,e_2$，…，$e_n$。

2. 由$\boldsymbol{x}_0$出发沿$\boldsymbol{p}_1$方向进行一维寻查，求得极小点$\boldsymbol{x}_1$；再由$\boldsymbol{x}_1$点出发沿$\boldsymbol{p}_2$方向进行一维寻查求得极小点$\boldsymbol{x}_2$；…。依次作下去，最后，由$\boldsymbol{x}_{n-1}$点出发沿$\boldsymbol{p}_n$方向求得极小点$\boldsymbol{x}_n$。

3. 在一维寻查过程中，用$\Phi_i=\Phi(\boldsymbol{x}_i)$表示点$\boldsymbol{x}_i$上的目标函数值，同时比较前后两次目标函数的差值，即比较以下序列

$$\Phi_0-\Phi_1,\ \Phi_1-\Phi_2,\ \cdots,\ \Phi_{n-1}-\Phi_n$$

的大小，用

$$\Delta_m=\Phi_{m-1}-\Phi_m \tag{2-5.12}$$

表示前后二次目标函数差值最大者。

4. 将$\boldsymbol{x}_0$到$\boldsymbol{x}_n$的距离延长一倍，即计算$2\boldsymbol{x}_n-\boldsymbol{x}_0$，在此新点上求目标函数值，表示为

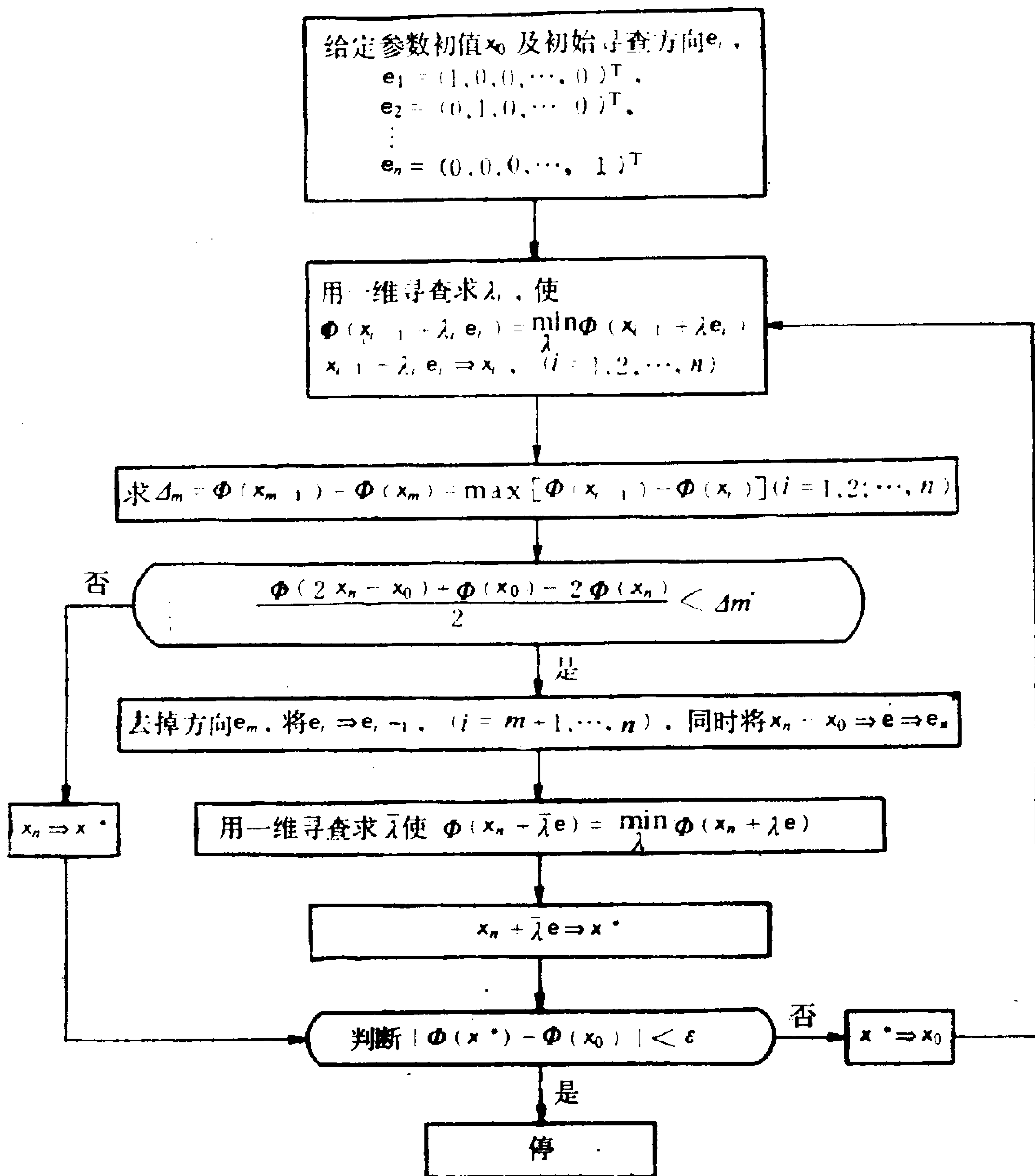
$$\tilde{\Phi}=\Phi(2\boldsymbol{x}_n-\boldsymbol{x}_0).$$

如果关系式

$$\frac{\Phi_0-2\Phi_n+\tilde{\Phi}}{2}\geqslant\Delta_m \tag{2-5.13}$$

成立，则表明沿$\boldsymbol{x}_n$延长方向寻查的目标函数极小点仍大于原方向各次迭代的目标函数的差值，说明这不是最好方向。故下一次迭代时，仍以$\boldsymbol{x}_n$为出发点，沿原方向$\boldsymbol{p}_1$，$\boldsymbol{p}_2$，…，$\boldsymbol{p}_n$进行寻查。

反之，若关系式



给定参数初值$x_0$及初始寻查方向$e_i$，
$e_1=(1,0,0,\cdots,0)^T$，
$e_2=(0,1,0,\cdots,0)^T$，
⋮
$e_n=(0,0,0,\cdots,1)^T$
用一维寻查求$\lambda_i$，使
$\Phi(x_{i-1}+\lambda_i e_i)=\min_{\lambda}\Phi(x_{i-1}+\lambda e_i)$
$x_{i-1}+\lambda_i e_i\Rightarrow x_i$，$(i=1,2,\cdots,n)$
求$\Delta_m=\Phi(x_{m-1})-\Phi(x_m)=\max[\Phi(x_{i-1})-\Phi(x_i)](i=1,2,\cdots,n)$
否
$\frac{\Phi(2x_n-x_0)+\Phi(x_0)-2\Phi(x_n)}{2}<\Delta m$
是
去掉方向$e_m$，将$e_i\Rightarrow e_{i-1}$，$(i=m+1,\cdots,n)$，同时将$x_n-x_0\Rightarrow e\Rightarrow e_n$
$x_n\Rightarrow x^*$
用一维寻查求$\bar{\lambda}$使 $\Phi(x_n+\bar{\lambda}e)=\min_{\lambda}\Phi(x_n+\lambda e)$
$x_n+\bar{\lambda}e\Rightarrow x^*$
判断 $|\Phi(x^*)-\Phi(x_0)|<\varepsilon$
否
$x^*\Rightarrow x_0$
是
停


图 2-5-2

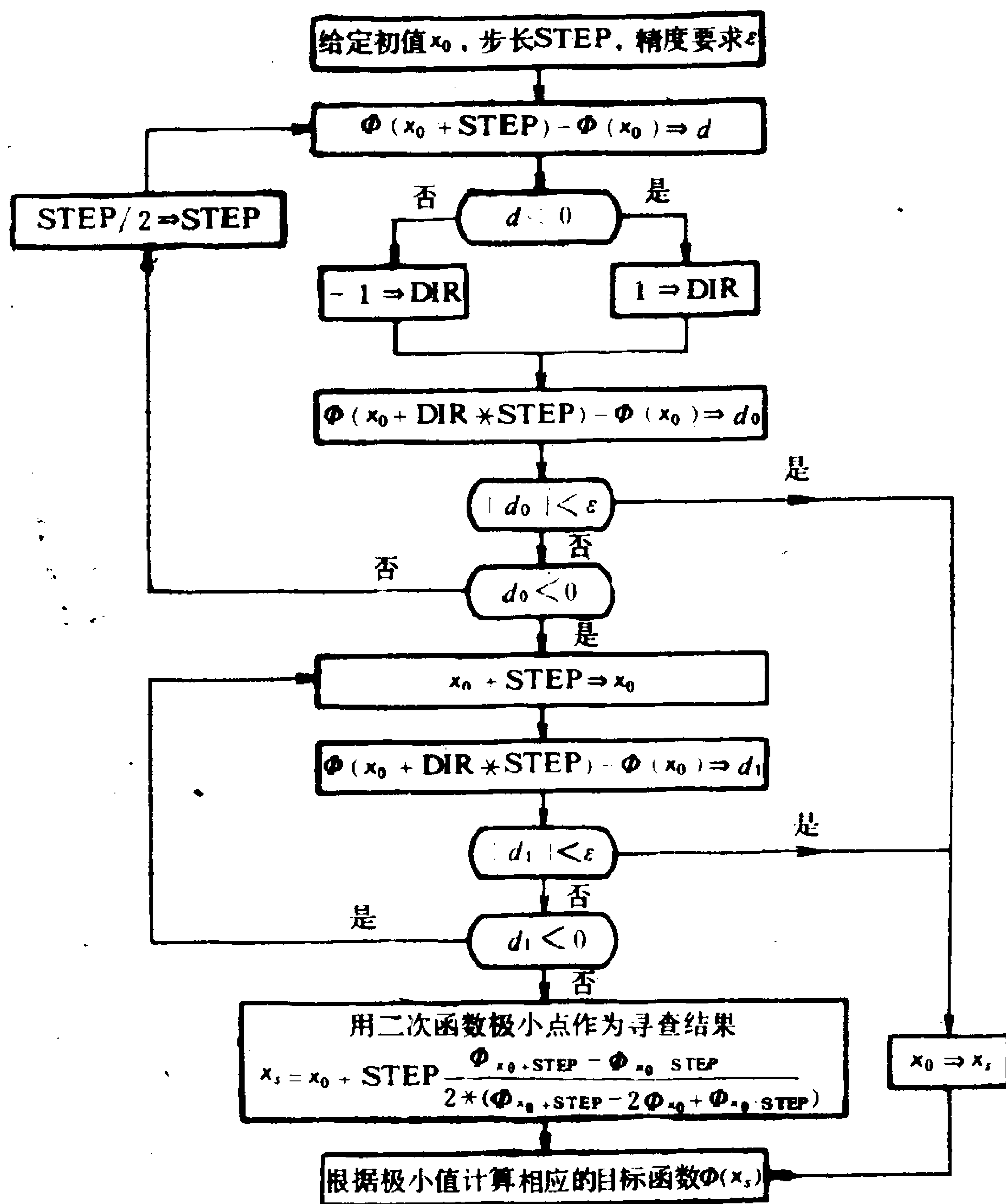

给定初值$x_0$，步长STEP，精度要求$\varepsilon$
$\Phi(x_0+STEP)-\Phi(x_0)\Rightarrow d$
STEP/2⇒STEP
否
是
$d<0$
-1⇒DIR
1⇒DIR
$\Phi(x_0+DIR*STEP)-\Phi(x_0)\Rightarrow d_0$
是
$|d_0|<\varepsilon$
否
否
$d_0<0$
是
$x_0+STEP\Rightarrow x_0$
$\Phi(x_0+DIR*STEP)-\Phi(x_0)\Rightarrow d_1$
是
$|d_1|<\varepsilon$
否
是
$d_1<0$
否
用二次函数极小点作为寻查结果
$x_s=x_0+STEP\dfrac{\Phi_{x_0+STEP}-\Phi_{x_0-STEP}}{2*(\Phi_{x_0+STEP}-2\Phi_{x_0}+\Phi_{x_0-STEP})}$
$x_0\Rightarrow x_s$
根据极小值计算相应的目标函数$\Phi(x_s)$

图 2-5-3

$$\frac{\Phi_0 - 2\Phi_n + \widetilde{\Phi}}{2} < \Delta_m \qquad (2\text{-}5.14)$$

成立，则用 $p$ 表示 $x_0$ 与 $x_n$ 联线方向

$$p = x_n - x_0. \qquad (2\text{-}5.15)$$

以 $x^*$ 表示沿此方向进行一维寻查求得的极小点，并将 $x^*$ 作为下次迭代的出发点，去掉 $p_m$ 方向，以 $p$ 方向加入组成新的方向组 $p_1$，$p_2$，…，$p_{m-1}$，$p_{m+1}$，…，$p_n$，$p$。用新方向组作为下次迭代单方向探测求极小的方向。

5．重复以上步骤，直到求得的 $x^*$ 点的目标函数值达到预定的精度，即前后两次迭代的目标函数值达到下式要求为止，

$$|\Phi(x^*) - \Phi(x_0)| < \varepsilon. \qquad (2\text{-}5.16)$$

图2-5-2为Powell算法计算框图。

在Powell法中要反复进行一维寻查，即沿某一方向求极小值。为此利用专门的子程序实现这一算法。一维寻查计算包括确定方向(*DIR*)，缩小极小值所在区间和用二次函数估计极小值等主要步骤。其计算框图如图2-5-3示。

利用 Powell 方法的迭代过程，对于二维情况可由图2-5-4表示。图中等值线为目标函数空间曲面在平面上的投影，变量 $x$ 的下角标表示迭代次数，上角标表示迭代时所循的方向。整个寻查过程由实线箭头所示。

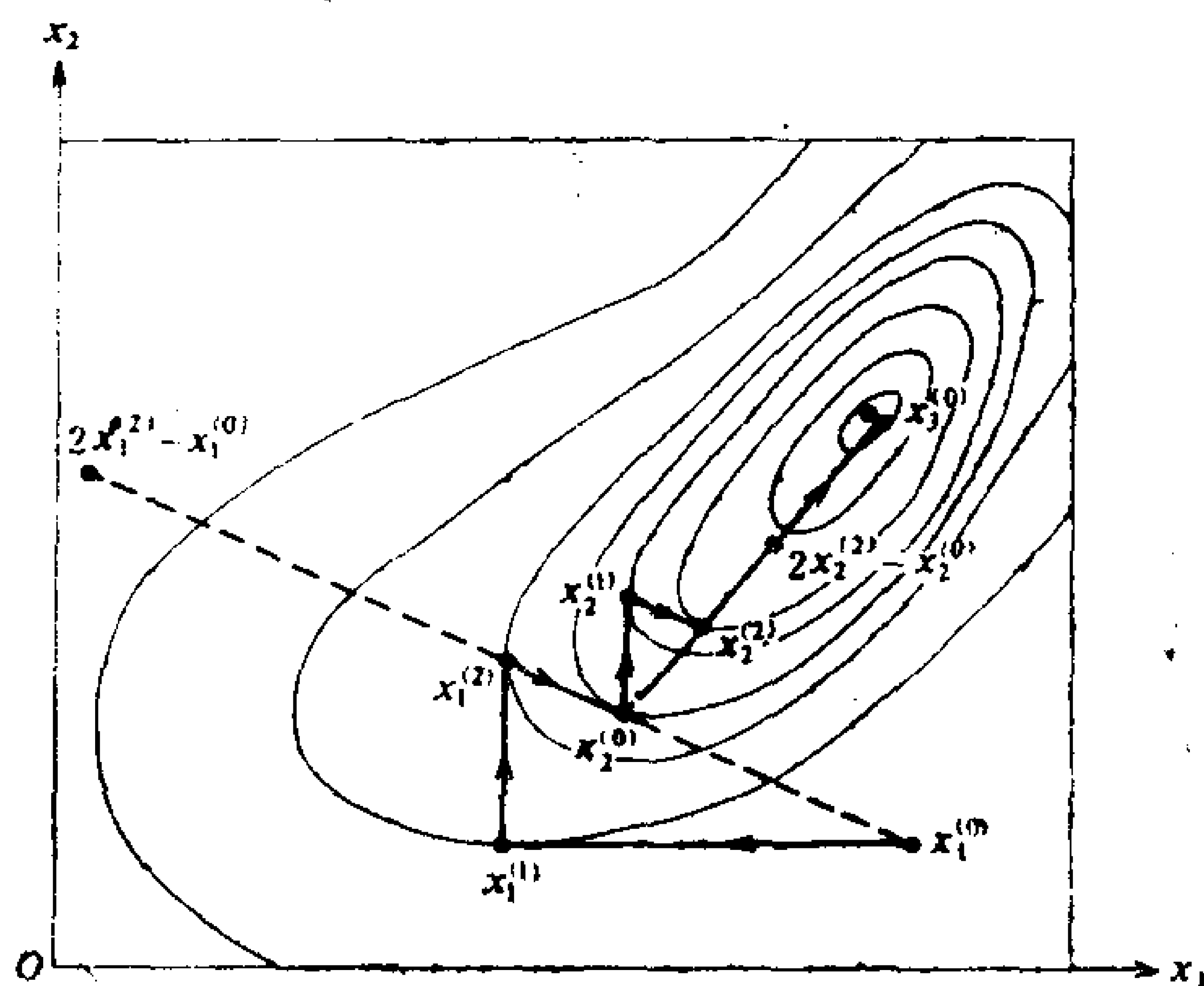



图 2-5-4

## §2-6 计算目标函数一阶导数的方法

目标函数$\Phi(\boldsymbol{x})$的一阶导数，就是函数的梯度。一阶导数的最优化问题的解法，通常可以分为两类：一类是直接用一阶导数来寻找极小点，如梯度法、共轭梯度法等，另一类是用一阶导数矩阵建立线性方程组，求解方程，来寻找极小点，这类如最小二乘法、阻尼最小二乘法等。

### （一）梯 度 法

梯度法又称最速下降法，或"瞎子下山"法。其基本思想是在初始点$\boldsymbol{x}_0$附近找一个下降最快的方向，即负梯度方向。沿负梯度方向找到一个下降最低的点，也就是沿此方向求单方向的极小点。以这点作为新的$\boldsymbol{x}_0$点。继续以上步骤进行寻查，逐渐逼近极小点。

将目标函数在$\boldsymbol{x}_0$附近展开，并略去二次及二次以上高次项，

$$\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})\cong\Phi(\boldsymbol{x}_0)+\sum_{i=1}^{n}\left.\frac{\partial\Phi(\boldsymbol{x})}{\partial x_i}\right|_{\boldsymbol{x}_0}\Delta x_i$$

$$=\Phi(\boldsymbol{x}_0)+\left.\frac{\partial\Phi}{\partial x_1}\right|_{\boldsymbol{x}_0}\Delta x_1+\left.\frac{\partial\Phi}{\partial x_2}\right|_{\boldsymbol{x}_0}\Delta x_2+\cdots+\left.\frac{\partial\Phi}{\partial x_n}\right|_{\boldsymbol{x}_0}\Delta x_n. \tag{2-6.1}$$

要找到一个下降点，即要求$\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})<\Phi(\boldsymbol{x}_0)$。为此，要求

$$\Phi(\boldsymbol{x}_0)-\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})>0, \tag{2-6.2}$$

且二者差别愈大愈好。

将(2-6.1)式代入(2-6.2)式有

$$\Phi(\boldsymbol{x}_0)-\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})$$

$$=-\left.\left(\frac{\partial\Phi}{\partial x_1}\Delta x_1+\frac{\partial\Phi}{\partial x_2}\Delta x_2+\cdots+\frac{\partial\Phi}{\partial x_r}\Delta x_n\right)\right|_{\boldsymbol{x}_0}$$

$$= -\left(\frac{\partial \Phi}{\partial x_1}, \frac{\partial \Phi}{\partial x_2}, \cdots, \frac{\partial \Phi}{\partial x_n}\right)_{\boldsymbol{x}_0} \cdot (\Delta x_1, \Delta x_2, \cdots, \Delta x_n)^T$$

$$= -g(\boldsymbol{x}_0) \cdot \Delta \boldsymbol{x}^T. \tag{2-6.3}$$

式中$g(\boldsymbol{x}_0) = \left(\frac{\partial \Phi}{\partial x_1}, \frac{\partial \Phi}{\partial x_2}, \cdots, \frac{\partial \Phi}{\partial x_n}\right)^T_{\boldsymbol{x}_0}$是目标函数在$\boldsymbol{x}_0$点的一阶导数各分量，称为目标函数的梯度或斜量。显然，要使(2-6.3)式有最大值，则应使点积

$$-g(\boldsymbol{x}_0) \cdot \Delta \boldsymbol{x} = -\|g\|_{\boldsymbol{x}_0} \|\Delta \boldsymbol{x}\| \cos\theta \tag{2-6.4}$$

有最大值。其中$\theta$为$g$与$\Delta \boldsymbol{x}$之夹角。显然，只有$\theta = \pi$，即进行寻查极小值方向$\Delta \boldsymbol{x}$与梯度方向$g$相反时，才能保证最快下降。沿负梯度方向进行单方向一维寻查，求得下降最低的点

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 - \lambda g. \tag{2-6.5}$$

其中$\lambda$为最优步长。然后又以$\boldsymbol{x}_1$为出发点，沿$\boldsymbol{x}_1$点负梯度方向寻查。如此反复迭代，直到目标函数$\Phi(\boldsymbol{x})$值达到预定精度要求为止。

这种方法仅着眼于初始点附近最速下降方向求极小。从整体来看，它可能是迂回前进的。故收敛速度慢，计算时间长。但它每迭代一步，都能保证$\Phi(\boldsymbol{x})$值下降，即能确保收敛，直至达到极小点。

## （二） 最小二乘法

对于地球物理反演问题，其目标函数一般都具有平方和形式

$$\Phi(\boldsymbol{x}) = \sum_{i=1}^{m} (y_i^0 - y_i(\boldsymbol{x}))^2. \tag{2-6.6}$$

式中$y_i^0$为观测值，$y_i(\boldsymbol{x})$为理论计算值。当$y_i(\boldsymbol{x})$为参数$\boldsymbol{x}$的线性函数时，可以直接化为线性方程组求解。当$y_i(\boldsymbol{x})$为非线性函数时，首先将$y_i(\boldsymbol{x})$在$\boldsymbol{x}_0$（初始点）附近线性展开，

$$y_i^c(\boldsymbol{x}_0+\Delta\boldsymbol{x})\cong y_i^c(\boldsymbol{x}_0)+\sum_{j=1}^{n}\left.\frac{\partial y_i^c}{\partial x_j}\right|_{\boldsymbol{x}_0}\Delta x_j,$$

$$(i=1,\ 2,\ \cdots,\ m). \qquad (2\text{-}6.7)$$

将上式写为矩阵形式即为

$$\boldsymbol{y}^c(\boldsymbol{x}_0+\Delta\boldsymbol{x})\cong\boldsymbol{y}^c(\boldsymbol{x}_0)+\boldsymbol{A}\Delta\boldsymbol{x}, \qquad (2\text{-}6.8)$$

式中$\boldsymbol{A}$为微分系数矩阵(或Jacobi阵)

$$\boldsymbol{A}_{m\times n}=\left(\begin{matrix}\frac{\partial y_1^c}{\partial x_1} & \frac{\partial y_1^c}{\partial x_2} & \cdots & \frac{\partial y_1^c}{\partial x_n}\\ \frac{\partial y_2^c}{\partial x_1} & \frac{\partial y_2^c}{\partial x_2} & \cdots & \frac{\partial y_2^c}{\partial x_n}\\ \vdots & \vdots & & \\ \frac{\partial y_m^c}{\partial x_1} & \frac{\partial y_m^c}{\partial x_2} & \cdots & \frac{\partial y_m^c}{\partial x_n}\end{matrix}\right)_{\boldsymbol{x}=\boldsymbol{x}_0}. \qquad (2\text{-}6.9)$$

而$\Delta\boldsymbol{x}$称为参数$\boldsymbol{x}$的改正值

$$\Delta\boldsymbol{x}=(\Delta x_1,\ \Delta x_2,\ \cdots,\ \Delta x_n)^T$$
$$=(x_1-x_0,\ x_2-x_0,\ \cdots,\ x_n-x_0)^T. \qquad (2\text{-}6.10)$$

将(2-6.7)式代入(2-6.6)式，并写为向量形式，得到$\boldsymbol{y}^c(\boldsymbol{x})$线性化后的目标函数估计值

$$\hat{\Phi}(\boldsymbol{x}_0+\Delta\boldsymbol{x})=[(\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0))-\boldsymbol{A}\Delta\boldsymbol{x}]^T[(\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0))$$
$$-\boldsymbol{A}\Delta\boldsymbol{x}]=[\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0)]^T[\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0)]$$
$$-2[\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0)]^T\boldsymbol{A}\Delta\boldsymbol{x}+\Delta\boldsymbol{x}^T\boldsymbol{A}^T\boldsymbol{A}\Delta\boldsymbol{x}.$$

令上式取极小值必要条件，利用矩阵微分公式有

$$\frac{\partial\hat{\Phi}}{\partial(\Delta\boldsymbol{x})}=-2\boldsymbol{A}^T[\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0)]+2\boldsymbol{A}^T\boldsymbol{A}\Delta\boldsymbol{x}=0.$$

由此得

$$\boldsymbol{A}^T\boldsymbol{A}\Delta\boldsymbol{x}=\boldsymbol{A}^T[\boldsymbol{y}^0-\boldsymbol{y}^c(\boldsymbol{x}_0)]. \qquad (2\text{-}6.11)$$

令上式右端项用$\boldsymbol{g}$表示

$$g=A^T[y^0-y^c(x_0)].\tag{2-6.12}$$

注意，平方和形式的目标函数的梯度为

$$\frac{\partial\Phi}{\partial x}=\left(\frac{\partial\Phi}{\partial x_1},\ \frac{\partial\Phi}{\partial x_2},\ \cdots,\ \frac{\partial\Phi}{\partial x_n}\right)^T$$

$$=-2\left[\sum_{i=1}^{m}\frac{\partial y_i^c}{\partial x_1}(y_i^0-y_i^c(x_0)),\sum_{i=1}^{m}\frac{\partial y_i^c}{\partial x_2}(y_i^0-y_i^c(x_0)),\right.$$

$$\left.\cdots,\ \sum_{i=1}^{m}\frac{\partial y_i^c}{\partial x_n}(y_i^0-y_i^c(x_0))\right]^T=-2A^T[y^0-y^c(x_0)].$$

$$\tag{2-6.13}$$

比较(2-6.12)式及(2-6.13)式，故(2-6.11)式中的右端项$g$是目标函数负梯度向量的一半

$$g=\frac{1}{2}\left(-\frac{\partial\Phi}{\partial x}\right),\tag{2-6.14}$$

由(2-6.11)式可以求出改正量(或增量)

$$\Delta x=(A^TA)^{-1}g.\tag{2-6.15}$$

由此可见，最小二乘法的校正方向为$(A^TA)^{-1}g$，而不同于负梯度方向。

对于实际问题，当已知观测数据和选定初始模型$x_0$后，Jacobi 矩阵$A$及右端项$g$都可以计算，由(2-6.15)式解出$\Delta x$，从而取

$$x_1=x_0+\Delta x\tag{2-6.16}$$

作为首次近似值，然后以$x_1$作为新的初值，继续迭代求得$x_2$，$x_3$，$\cdots x_k$，$\cdots$，直到目标函数差值$\Phi(x_n)-\Phi(x_{n-1})$达到预定精度要求为止。

以上方法就是Gauss 于 1809年提出的最小二乘法(或 Gauss 法)。应用最小二乘法时，要求方阵$A^TA$为非奇异且正定的矩阵，否则无法区别极大点、极小点与鞍点，或者不能求解。一般

说来，Gauss 法与初值$x_0$的选取有密切关系。如果选择不当，迭代往往失败。

为了提高计算效果，可以采用下述一维寻查的办法。在求出改正量$\Delta x$后，不直接用$x_0+\Delta x$作为解的第一次近似，而仅把$\Delta x$当作寻查方向。即从$x_0$点出发，沿$\Delta x$方向进行一维寻查求$\Phi(x)$的极小值

$$\Phi(x_0+\lambda_0\Delta x)=\min_{\lambda}\Phi(x_0+\lambda\Delta x), \tag{2-6.17}$$

$\lambda_0$称为最优步长因子，我们取

$$x_1=x_0+\lambda_0\Delta x \tag{2-6.18}$$

作为第一次近似值。然后依次迭代，取

$$x_{k+1}=x_k+\lambda_k\Delta x_k,\ k=1,\ 2,\ \cdots. \tag{2-6.19}$$

直到得出满足精度要求的解为止。我们称此算法为广义最小二乘法。这种改进的最小二乘法，在多数情况下可以保证收敛，但有时前进得很慢，很费计算时间。

### （三）阻尼最小二乘法（Marquardt方法）

为了改善Gauss 法迭代过程发散的问题，1944年Levenberg及1963年Marquardt先后提出了改进办法，这就是目前普遍认为效果较好并得到广泛运用的阻尼最小二乘法。

我们知道，用最小二乘法进行迭代求解时收敛性很不稳定，易于造成发散。对于二次函数来说，最小二乘法校正向量的方向直接指向极小点$x^*$，与最速下降法负梯度方向并不一致。当目标函数$\Phi(x)$呈拉长的椭球面时，两种方法前进的方向之间夹角甚至可达80°—90°。

阻尼最小二乘法就是在最小二乘法与最速下降法之间采取某种插值，要求增量$\Delta x$在保证目标函数$\Phi(x)$在下降的方向上以最大步长前进。它的基本思想可描述如下，在迭代的每一步，最好尽量使用接近最小二乘改正方向，以便使校正向量的步长尽可能地增大。如果这种情况不能收敛时，再逐渐改用接近最速下降法

的负梯度方向，同时缩小校正步长以保证收敛。为实现这种想法，只须将式(2-6.11)中正规方程组的系数矩阵$A^TA$的对角元素适当加大，也就是用$A^TA+\nu^2I$代替$A^TA$。这里$I$是单位矩阵，$\nu$为适当的正数，称为阻尼因子。这样便可保证正规方程组总是非奇异正定的。于是(2-6.11)原式$A^TA\Delta x=g$就可改写为

$$(A^TA+\nu^2I)\Delta x=g, \qquad (2\text{-}6.20)$$

校正向量的解为

$$\Delta x=(A^TA+\nu^2I)^{-1}g. \qquad (2\text{-}6.21)$$

由式(2-6.21)不难看出，迭代过程中阻尼系数$\nu$起着重大作用。当$\nu=0$时，退化为一般最小二乘法。当由零渐逐增大时，$(A^TA+\nu^2I)$的对角元素也随之增大，解$\Delta x$逐渐由最小二乘方向转向最速下降方向$g$。$\nu$愈大，愈接近于最速下降方向。另一方面，当$\nu$愈来愈大时，校正向量$\Delta x$长度也愈来愈小，以至趋于零。这种情况下，虽然收敛很慢，但总能保证迭代过程稳定地进行。

在迭代过程中，阻尼系数是变化的。对于第$i$次迭代，对给定的阻尼系数$\nu_i$解出的校正向量为$\Delta x_i$。如果新解不能使目标函数下降，即出现

$$\Phi(x_i+\Delta x_i)\geqslant\Phi(x_i)$$

的情况，这说明迭代出现发散。这时，应加大阻尼系数，根据计算实践，一般可增大1.5倍，即取

$$\nu_{i+1}=1.5\nu_i, \qquad (2\text{-}6.22)$$

再代入式(2-6.21)重新求解。反之，如第$i$次迭代使目标函数下降，

$$\Phi(x_i+\Delta x_i)<\Phi(x_i).$$

说明校正步长还可以加大，即阻尼系数$\nu$还有可能再缩小一些。在下一次迭代中，可以把$\nu$值再缩小一半，即

$$\nu_{i+1}=0.5\nu_i. \qquad (2\text{-}6.23)$$

式中$\nu_i$的初值可以选在1—10之间。

与Gauss法类似，为了提高本方法的计算效果，往往采用一

维寻查法进行修定。

在求得 $\Delta \boldsymbol{x}$ 以后，并不直接用它来修定参数$\boldsymbol{x}$，而是以它作为寻查方向，沿此方向进行一维寻查求极小点。其具体作法如下。

对于一维寻查函数

$$\Phi(\boldsymbol{x}_0+\lambda_0\Delta\boldsymbol{x})=\min_{0\leqslant\lambda\leqslant1}\Phi(\boldsymbol{x}_0+\lambda\Delta\boldsymbol{x}).$$

先取$\lambda=1$，即计算$\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})$，若

$$\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})<\Phi(\boldsymbol{x}_0),$$

则认为新的极小点近似解$\boldsymbol{x}_1=\boldsymbol{x}_0+\Delta\boldsymbol{x}$已经得到。否则，利用$\Phi(\boldsymbol{x}_0+\lambda\Delta\boldsymbol{x})$在$\lambda=0$，1 上的函数值$\Phi_0$及$\Phi_1$，以及$\lambda=0$时$\Phi$对$\lambda$的导数$\Phi'(\boldsymbol{x}_0)$作抛物线插值。用抛物线的极小点坐标$\lambda^*$代入求极小点$\boldsymbol{x}^*=\boldsymbol{x}_0+\lambda^*\Delta\boldsymbol{x}$。

抛物线法求极小点坐标$\lambda^*$的计算公式推导如下，如图 2-6-1 示，设目标函数自变量$\lambda$在〔0,1〕区间内为二次函数关系

$$\Phi(\boldsymbol{x}_0+\lambda\Delta\boldsymbol{x})=a_0+a_1\lambda+a_2\lambda^2. \tag{2-6.24}$$

故

$$\left.\begin{aligned}&\Phi(\boldsymbol{x}_0)=a_0\quad(\lambda=0),\\&\Phi(\boldsymbol{x}_0+\Delta\boldsymbol{x})=a_0+a_1+a_2\quad(\lambda=1),\end{aligned}\right\} \tag{2-6.25}$$

以及

$$\Phi'(\boldsymbol{x}_0)=\left.\frac{d\Phi}{d\lambda}\right|_{\lambda=0}=a_1. \tag{2-6.26}$$

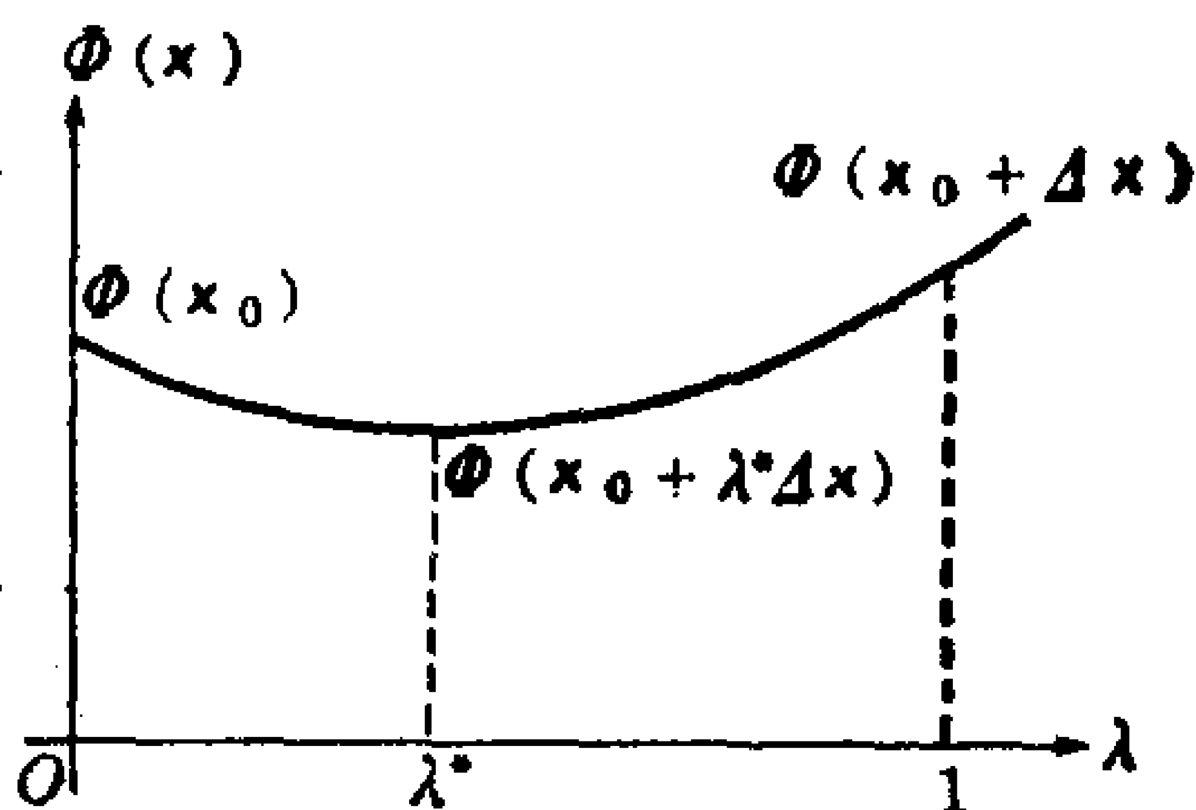



图 2-6-1

在抛物线顶点有极小值$\lambda^*$，由此条件得

$$\Phi'(\boldsymbol{x}_0+\lambda^*\Delta\boldsymbol{x})=a_1+2a_2\lambda^*=0,$$

可以解出

$$\lambda^*=-\frac{a_1}{2a_2}. \tag{2-6.27}$$

由(2-6.25)及(2-6.26)

式不难得出

$$a_2 = \Phi(\boldsymbol{x}_0 + \Delta\boldsymbol{x}) - \Phi(\boldsymbol{x}_0) - \Phi'(\boldsymbol{x}_0). \qquad (2\text{-}6.28)$$

将(2-6.26)及(2-6.28)式代入(2-6.27)式不难得出

$$\lambda^* = \frac{-\Phi'(\boldsymbol{x}_0)}{2[\Phi(\boldsymbol{x}_0 + \Delta\boldsymbol{x}) - \Phi(\boldsymbol{x}_0) - \Phi'(\boldsymbol{x}_0)]}. \qquad (2\text{-}6.29)$$

对于平方和形式的目标函数，由(2-6.13)式可导出以下关系

$$\Phi'(\boldsymbol{x}_0) = -2\sum_{i=1}^{n} g_i \Delta x_i. \qquad (2\text{-}6.30)$$

式中$g_i$为目标函数负梯度$\boldsymbol{g} = \boldsymbol{A}^T[\boldsymbol{y}^0 - \boldsymbol{y}^c(\boldsymbol{x}_0)]$的各分量，$\Delta x_i$为校正向量$\Delta\boldsymbol{x}$的各分量。把(2-6.30)代入(2-6.29)式，即得

$$\lambda^* = \frac{\sum\limits_{i=1}^{n} g_i \Delta x_i}{\left[\Phi(\boldsymbol{x}_0 + \Delta\boldsymbol{x}) - \Phi(\boldsymbol{x}_0) + 2\sum\limits_{i=1}^{n} g_i \Delta x_i\right]}. \qquad (2\text{-}6.31)$$

上式就是用抛物线法求极小点最优步长 $\lambda^*$ 的一般公式。由于式中含梯度项$g_i$，在一阶导数计算中是方便的。

若 $\Phi(\boldsymbol{x}_0 + \lambda^*\Delta\boldsymbol{x}) < \Phi(\boldsymbol{x}_0)$，则认为新的近似解 $\boldsymbol{x}_1 = \boldsymbol{x}_0 + \lambda^*\Delta\boldsymbol{x}$ 已经得到。如果目标函数还不下降，则可利用下述二分法。

分别取 $t = \frac{\lambda^*}{2}, \frac{\lambda^*}{2^2}, \cdots, \frac{\lambda^*}{2^7}$，并检验 $\Phi(\boldsymbol{x}_0 + t\Delta\boldsymbol{x})$ 是否小于 $\Phi(\boldsymbol{x}_0)$。一旦其中某一$t$值使 $\Phi(\boldsymbol{x}_0 + t\Delta\boldsymbol{x}) < \Phi(\boldsymbol{x}_0)$，就取新近似解为$\boldsymbol{x}_1 = \boldsymbol{x}_0 + t\Delta\boldsymbol{x}$。最多二分7次，若7次之后函数仍不下降，则认为 $\boldsymbol{x}_0$ 已是$\Phi(\boldsymbol{x})$的极小值点。

求出新的近似解 $\boldsymbol{x}_1$后，重复上述步骤依次迭代求出$\boldsymbol{x}_2, \boldsymbol{x}_3, \cdots$，直到目标函数$\Phi(\boldsymbol{x})$迭代前后两次之差满足预定精度要求，即$\Phi(\boldsymbol{x}) - \Phi(\boldsymbol{x} + \Delta\boldsymbol{x}) < \varepsilon$，或修改量$\Delta\boldsymbol{x} < \delta$为止。

在用一阶导数方法计算最优化问题时，都要涉及到Jacobi矩阵式(2-6.9)的形成问题。由于理论计算值$y^c(\boldsymbol{x})$多数情况下不能用简单函数表示或者根本写不出函数形式。因此，无法用函数直接微分形式进行计算。经常采用的是扰动法。即给自变量一系列微小的变化，利用有限差商格式分别计算 Jacobi 矩阵中各元素。设对自变量$\boldsymbol{x}$的扰动为

$$\delta \boldsymbol{x} = (\delta x_1,\ \delta x_2,\ \cdots,\ \delta x_n)^T. \qquad (2\text{-}6.32)$$

研究函数$y^c$中各个分量$y_1^c$，$y_2^c$，…，$y_m^c$如何变化，并采用以下有限差商公式计算，

$$\frac{\partial y_i^c(\boldsymbol{x})}{\partial x_j} = \frac{y_i^c(x_1, x_2, \cdots, x_j + \delta x_j, \cdots, x_n) - y_i^c(x_1, x_2, \cdots, x_j, \cdots x_n)}{\delta x_j},$$

$$(i=1,\ 2,\ \cdots,\ m;\ j=1,\ 2,\ \cdots,\ n). \qquad (2\text{-}6.33)$$

通常取$\delta x_1 = \delta x_2 = \cdots = \delta x_n$为$10^{-5}$或$10^{-2}$，视具体情况而定。

## §2-7 计算目标函数二阶导数的方法

现在我们来考虑把适用于平方和形式目标函数的最小二乘法的思想推广到一般的目标函数上去，以建立另一类寻查方向序列。这类方法的主要特点是利用目标函数的二阶导数值，或者不直接计算二阶导数而用近似方法逐步获得二阶导数，以得到目标函数的极小点。以下介绍拟 Newton 法。

设$\Phi(\boldsymbol{x})$为一般的目标函数，具有连续的一、二阶偏导数。对给定点$\boldsymbol{x}_i$附近对$\Phi(\boldsymbol{x})$作二次Taylor多项式逼近

$$\Phi(\boldsymbol{x}) \cong \Phi(\boldsymbol{x}_i) + \boldsymbol{g}_i^T \Delta \boldsymbol{x} + \frac{1}{2} \Delta \boldsymbol{x}^T \boldsymbol{G}_i \Delta \boldsymbol{x}. \qquad (2\text{-}7.1)$$

式中$\boldsymbol{g}_i = \nabla \Phi(\boldsymbol{x}_i)$，为目标函数在$\boldsymbol{x}_i$点的梯度向量（或斜量），$\boldsymbol{G}_i$为$\Phi(\boldsymbol{x})$在$\boldsymbol{x}_i$点的二阶导数矩阵（或Hesse 矩阵），

$$
G_{n\times n}=\begin{bmatrix}
\frac{\partial^2\Phi}{\partial x_1^2} & \frac{\partial^2\Phi}{\partial x_1\partial x_2} & \frac{\partial^2\Phi}{\partial x_1\partial x_3} & \cdots & \frac{\partial^2\Phi}{\partial x_1\partial x_n}\\
\frac{\partial^2\Phi}{\partial x_2\partial x_1} & \frac{\partial^2\Phi}{\partial x_2^2} & \frac{\partial^2\Phi}{\partial x_2\partial x_3} & \cdots & \frac{\partial^2\Phi}{\partial x_2\partial x_n}\\
\frac{\partial^2\Phi}{\partial x_3\partial x_1} & \frac{\partial^2\Phi}{\partial x_3\partial x_2} & \frac{\partial^2\Phi}{\partial x_3^2} & \cdots & \frac{\partial^2\Phi}{\partial x_3\partial x_n}\\
\vdots & \vdots & \vdots & & \vdots\\
\frac{\partial^2\Phi}{\partial x_n\partial x_1} & \frac{\partial^2\Phi}{\partial x_n\partial x_2} & \frac{\partial^2\Phi}{\partial x_n\partial x_3} & \cdots & \frac{\partial^2\Phi}{\partial x_n^2}
\end{bmatrix}_{\boldsymbol{x}=\boldsymbol{x}_i}.
$$

对式(2-7.1)求目标函数梯度

$$\nabla\Phi(\boldsymbol{x})\cong\boldsymbol{g}_i+\boldsymbol{G}_i\Delta\boldsymbol{x}, \tag{2-7.2}$$

在极小点$\nabla\Phi(\boldsymbol{x})=0$，于是(2-7.2)式可以简化为

$$\boldsymbol{G}_i\Delta\boldsymbol{x}=-\boldsymbol{g}_i.$$

由此得改正向量

$$\Delta\boldsymbol{x}=-\boldsymbol{G}_i^{-1}\boldsymbol{g}_i. \tag{2-7.3}$$

如果$\Phi(\boldsymbol{x})$本身就是二次函数，那么$\boldsymbol{G}_i$是一个常矩阵，其元素为常量，这时(2-7.1)式是准确的，因此，从任何一点$\boldsymbol{x}_i$出发，用公式(2-7.3)一次使可求得较正量$\Delta\boldsymbol{x}$，达到$\Phi(\boldsymbol{x})$的极小点$\boldsymbol{x}^*=\boldsymbol{x}_i+\Delta\boldsymbol{x}$。

当$\Phi(\boldsymbol{x})$不是二次函数时，(2-7.1)仅是目标函数在$\boldsymbol{x}_i$点附近的近似表达式。由(2-7.3)求得的$\Delta\boldsymbol{x}$，也不能一次达到极小点。因此，常取

$$\boldsymbol{p}_i=-\boldsymbol{G}_i^{-1}\boldsymbol{g}_i \tag{2-7.4}$$

为寻查方向，沿此方向作一维寻查，即求$\lambda_i$使

$$\Phi(\boldsymbol{x}_i+\lambda_i\boldsymbol{p}_i)=\min_{\lambda}\Phi(\boldsymbol{x}_i+\lambda\boldsymbol{p}_i). \tag{2-7.5}$$

然后令$\boldsymbol{x}_{i+1}=\boldsymbol{x}_i+\lambda_i\boldsymbol{p}_i$，依次迭代，即可得极小点。

按照这种方式求出函数$\Phi(\boldsymbol{x})$极小点的方法，叫做Newton法，$\boldsymbol{p}_i$称为$\boldsymbol{x}_i$点的Newton方向。如果目标函数$\Phi(\boldsymbol{x})$的二阶导数便于计算，上述方法的收敛速度是较快的。

但是，由于实际问题中计算目标函数的二次导数相当复杂，甚至是不可能的。另外，在计算寻查方向 $p_i$ 时还得求逆矩阵 $G_i^{-1}$，当 $n$ 较大时，计算工作量是很大的。

现在考虑不计算二阶导数矩阵 $G_i$ 及其求逆问题，而设法构造另一个矩阵 $H_i$，用它来直接逼近逆阵 $G_i^{-1}$，我们称这一类方法为拟Newton法。

为了研究 $G_i^{-1}$ 的近似表达式 $H_i$，先来分析一下这个矩阵和斜量之间的关系。假定目标函数 $\Phi(\boldsymbol{x})$ 本身就是一个二次函数，它的二阶导数矩阵 $G$ 是一个常矩阵。它在 $\boldsymbol{x}_i$ 及 $\boldsymbol{x}_{i+1}$ 两点的斜量分别为 $g_i$ 和 $g_{i+1}$，据(2-7.2)式，可以导出下述关系：

$$g_{i+1}-g_i=G(\boldsymbol{x}_{i+1}-\boldsymbol{x}\ ), \tag{2-7.6}$$

或者

$$\boldsymbol{x}_{i+1}-\boldsymbol{x}_i=G^{-1}(g_{i+1}-g_i). \tag{2-7.7}$$

因此，可以要求 $G^{-1}$ 的 $i+1$ 次近似矩阵 $H_{i+1}$ 满足关系，

$$H_{i+1}(g_{i+1}-g_i)=\boldsymbol{x}_{i+1}-\boldsymbol{x}_i. \tag{2-7.8}$$

分别令

$$y_i=g_{i+1}-g_i,\quad \sigma_i=\boldsymbol{x}_{i+1}-\boldsymbol{x}_i.$$

于是式(2-7.8)可以简写为

$$H_{i+1}y_i=\sigma_i. \tag{2-7.9}$$

这一式子就是近似矩阵 $H_{i+1}$ 所应满足的条件，或者说是构造 $H_{i+1}$ 的基础。由于公式中只包含斜量(一阶导数)，而不包含二阶导数，计算是比较容易的。

具体产生 $H_{i+1}$ 阵的一种计算式称DFP(Davidon-Fletcher-Powell)公式，

$$H_{i+1}=H_i+\sigma_i\sigma_i^T/\sigma_i^T y_i-H_i y_i y_i^T H_i/y_i^T H_i y_i. \tag{2-7.10}$$

由这种公式进行寻查极小点的方法称DFP变尺度法，其计算步骤如下。

给定初始点 $\boldsymbol{x}_0$，取 $H_0=I$，令 $p_0=-H_0 g_0$，从 $\boldsymbol{x}_0$ 出发，沿 $p_0$ 方向进行一维寻查求极小点。然后令 $\boldsymbol{x}_1=\boldsymbol{x}_0+\lambda_0 p_0$，并计算

梯度$g_1$。

利用式(2-7.10)，计算出$H_1$。下一步从$x_1$点出发沿$p_1=-H_1g_1$方向进行一维寻查求极小点$x_2$，然后沿$p_2=-H_2g_2$求$x_3$，……。依次进行迭代，就可算出各次相应的近似矩阵$H_i$，以及极小点$x_i$。直到满足给定精度要求为止。

产生近似矩阵$H_{i+1}$还有一个BFS (Broyden-Fletcher-Shanno)公式，其表达式为

$$H_{i+1}=H_i+(\mu_i\sigma_i\sigma_i^T-H_iy_i\sigma_i^T-\sigma_iy_i^TH_i)/\sigma_i^Ty_i. \tag{2-7.11}$$

式中 $\mu_i=1+y_i^TH_iy_i/\sigma_i^Ty_i$.

目前，普遍认为变尺度法(即拟Newton法)的收敛速度和迭代稳定性都是令人满意的方法。

最后，我们概括地讨论一下这几种无约束最优化方法的应用情况。

根据对地震学及重力、地磁反演问题的大量计算对比。在各类常用的最优化方法中，以阻尼最小二乘法收敛速度最快。但在迭代过程中不如拟Newton法和直接法稳定。拟Newton法（包括DFP和BFS公式)的收敛速度仅次于阻尼最小二乘法，但迭代过程稳定。特别是反演参数较多的情况下，它突出地优于阻尼最小二乘法。直接法各类方法一般说来稳定性很好，但当参数增多时，其收敛速度明显减慢，要化费大量计算时间。

## §2-8 广义反演线性方程组的修正

以上各节讨论线性反演的数值解时，都没有考虑到模型参数或观测数据的量纲，也没有考虑到数据的统计特性。实际上，我们不仅常常遇到待反演的参数量纲上有很大差别，而且会遇到不统一的，不独立的和不精确的数据的反演问题。

这时，如果仍按照线性方程组$Ax=b$直接用广义逆或最优

化方法去作反演计算，则一般得不到良好结果。为此，必须对线性方程组进行修正，以使方程组规一化。规一化主要有以下途径。

（1）用一个 $m\times m$ 矩阵 $\boldsymbol{G}$ 左乘矩阵 $\boldsymbol{A}$ 和向量 $\boldsymbol{b}$，这相当于对观测数据进行规一化。

（2）用一个 $n\times n$ 矩阵 $\boldsymbol{E}$ 右乘矩阵 $\boldsymbol{A}$，这相当于对参数 $\boldsymbol{x}$ 进行规一化。

（3）在矩阵 $\boldsymbol{A}$ 的每一主对角元素上加上一个数，这相当于阻尼最小二乘法。

下面分别讨论这几种修正办法。

## （一）用 $G$ 左乘 $A$ 及 $b$

这种运算可用来改变对各残差分量的权重。设线性方程组的残差向量为

$$\boldsymbol{\varepsilon}=\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}. \tag{2-8.1}$$

令残差平方和极小，

$$\boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon}=(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x})^T(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x})=\min. \tag{2-8.2}$$

为了对各残差分量 $\varepsilon_i$ 取不同的权重，用一个权矩阵

$$\boldsymbol{W}=\boldsymbol{G}^T\boldsymbol{G}, \tag{2-8.3}$$

对残差平方和进行变换，即求

$$\boldsymbol{\varepsilon}^T\boldsymbol{W}\boldsymbol{\varepsilon}=\min, \tag{2-8.4}$$

这相当于计算

$$(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x})^T\boldsymbol{G}^T\boldsymbol{G}(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x})=\min, \tag{2-8.5}$$

换言之，相当于用矩阵 $\boldsymbol{G}$ 左乘 $\boldsymbol{A}$ 及 $\boldsymbol{b}$，

$$(\boldsymbol{G}\boldsymbol{b}-\boldsymbol{G}\boldsymbol{A}\boldsymbol{x})^T(\boldsymbol{G}\boldsymbol{b}-\boldsymbol{G}\boldsymbol{A}\boldsymbol{x})=\min. \tag{2-8.6}$$

最常见的一种情况是取权系数矩阵 $\boldsymbol{G}$ 为对角矩阵形式，因此有

$$\|\boldsymbol{G}\boldsymbol{\varepsilon}\|=\sum_{i=1}^{m} g_{ii}^2\varepsilon_i^2=\sum_{i=1}^{m} w_{ii}\varepsilon_i^2. \tag{2-8.7}$$

式(2-8.7)的情况相当于方程组$\boldsymbol{Ax}=\boldsymbol{b}$的第$i$个方程赋以较大的权系数$|g_{ii}|$，或$w_{ii}$。这样就使得第$i$个残差分量$|\varepsilon_i|$的作用相对变小。因而，若某些观测数据$b_i$有较高的精度(误差较小)，则我们对第$i$个方程引入较大的权系数。最方便的是取权系数为观测值标准误差$\sigma_i$的倒数，

$$g_{ii}=\frac{1}{\sigma_i}, \tag{2-8.8a}$$

或

$$w_{ii}=\frac{1}{\sigma_i^2}. \tag{2-8.8b}$$

用这种加权的最小二乘法对观测值进行规一化运算，可使规一化后的向量$\tilde{\boldsymbol{b}}$的各分量

$$\tilde{\boldsymbol{b}}_i=g_{ii}b_i=\frac{b_i}{\sigma_i} \tag{2-8.9}$$

都具有单位标准差，即

$$\sigma_{\tilde{b}_i}=1.$$

更一般地，如果我们有关于$\boldsymbol{b}$的误差更为充分的统计信息，即知道$\boldsymbol{b}$的不同分量间误差的相关性。则可以将观测值的协方差矩阵$\boldsymbol{C}$写出来。对$m\times m$正定对称的协方差矩阵$\boldsymbol{C}$进行Cholesky分解，以获得下三角矩阵$\boldsymbol{F}$，使满足

$$\boldsymbol{C}=\boldsymbol{FF}^T. \tag{2-8.10}$$

于是加权矩阵$\boldsymbol{G}$可以定义为

$$\boldsymbol{G}=\boldsymbol{F}^{-1}. \tag{2-8.11}$$

## （二）以$\boldsymbol{E}$右乘$\boldsymbol{A}$及参量变换$\boldsymbol{X}=\boldsymbol{E}\tilde{\boldsymbol{X}}+\boldsymbol{\xi}$

对于线性方程组$\boldsymbol{Ax}\cong\boldsymbol{b}$可改用方程组

$$\tilde{\boldsymbol{A}}\boldsymbol{x}\cong\tilde{\boldsymbol{b}} \tag{2-8.12}$$

代替，其中

$$\widetilde{A}=AE, \tag{2-8.13}$$

$$\widetilde{b}=b-A\xi, \tag{2-8.14}$$

$$x=E\widetilde{x}+\xi. \tag{2-8.15}$$

其中$E$为$n\times n$阶矩阵，$\xi$为与$x$的同维向量。若进一步将$E$定义为对角矩阵，变换(2-8.13)实际上将是对$A$的列向量作加权乘积运算，也就是$A$的每一个列向量都乘上$E$某一对应的主对角元素。

若$E$为$n\times n$阶非奇异矩阵，这种变换在数学上并不改变问题的实质。换句话说，由残差范数$\|\widetilde{b}-\widetilde{A}\widetilde{x}\|$极小而得到的一组解$x=E\widetilde{x}+\xi$与使$\|b-Ax\|$极小的一组解$x$是类似的。

然而，除非$E$是正交矩阵，变换后$\widetilde{A}$的条件数一般与$A$是不同的。这将会引起衡量解向量的"大小"即范数$\|x\|$的改变。因此，取不同的变换矩阵$E$，就会导致不同的解向量以供选择"最短长度"解。在问题$Ax\cong b$中的最小长度解是使$\|x\|$取极小的解，而在变换后方程组$\widetilde{A}\widetilde{x}\cong\widetilde{b}$的最小长度解是使$\|\widetilde{x}\|$取极小的解。这意味着宁取$\|E^{-1}(x-\xi)\|$的极小而不取$\|x\|$的极小。

对矩阵$E$的选择是基于解向量$x$的先验知识。设$x$与已知向量$\xi$很接近(称$\xi$为$x$的先验期望值)。再设已知道的估计值$\xi_i$的先验标准偏差为$\sigma_i$，于是可以取$E$为$n\times n$对角矩阵，其对角元素为

$$e_{ii}=\sigma_i. \tag{2-8.16}$$

于是，经过变换参数$\widetilde{x}$的每一分量为

$$\widetilde{x}_i=\frac{x_i-\xi_i}{\sigma_i}. \tag{2-8.17}$$

它是一个具有单位先验标准误差和期望值为零的随机变量。这样，就将参数规一化了。用这种规一化的参数进行计算，就可求得最小范数解$\|\widetilde{x}\|$

如果事前不知道关于$E$的估计值，则也可按矩阵$\tilde{A}$的列向量的欧氏模作为权系数。即

$$e_{ii}=\begin{cases}\|a_i\|^{-1}, & 若\|a_i\|\neq 0;\\ 1, & 若\|a_i\|=0.\end{cases} \tag{2-8.18}$$

式中 $a_i$ 表示矩阵$A$的第$i$列向量。

为了求得最小范数最小二乘解，可以把用$G$左乘$A$及用$E$右乘$A$两种变换结合起来，对方程组$Ax\cong b$同时进行修整。

## （三）加阻尼因子

对于任意线性方程组$Ax\cong b$的广义逆解为(用奇异值分解)

$$x=VS^{-1}U^Tb. \tag{2-8.19}$$

显然，解向量$x$的某一分量$x_i$直接与对角矩阵$S^{-1}$的因子$1/S_i$有关。当某个本征值$S_i\to 0$时，解分量$x_i$便可能出现发散情况。如图 2-8-1 所示。为了克服这种困难，可以采用 Levenberg (1944)及 Marquardt(1963)等人提出的加阻尼因子的方法。即在对角矩阵$S^{-1}$每一项加一阻尼系数，即使$S_i\to 0$，也不致发散。经过修改后的对角矩阵为

$$(S^2+\nu^2I)^{-1}S=\begin{bmatrix}\frac{S_1}{S_1^2+\nu^2} & & & \\ & \frac{S_2}{S_2^2+\nu^2} & & \\ & & \ddots & \\ & & & \frac{S_n}{S_n^2+\nu^2}\end{bmatrix}. \tag{2-8.20}$$

于是解向量

$$x=V[(S^2+\nu^2I)^{-1}S]U^Tb. \tag{2-8.21}$$

由阻尼最小二乘法建立的计算公式性质如下，当某一本征值

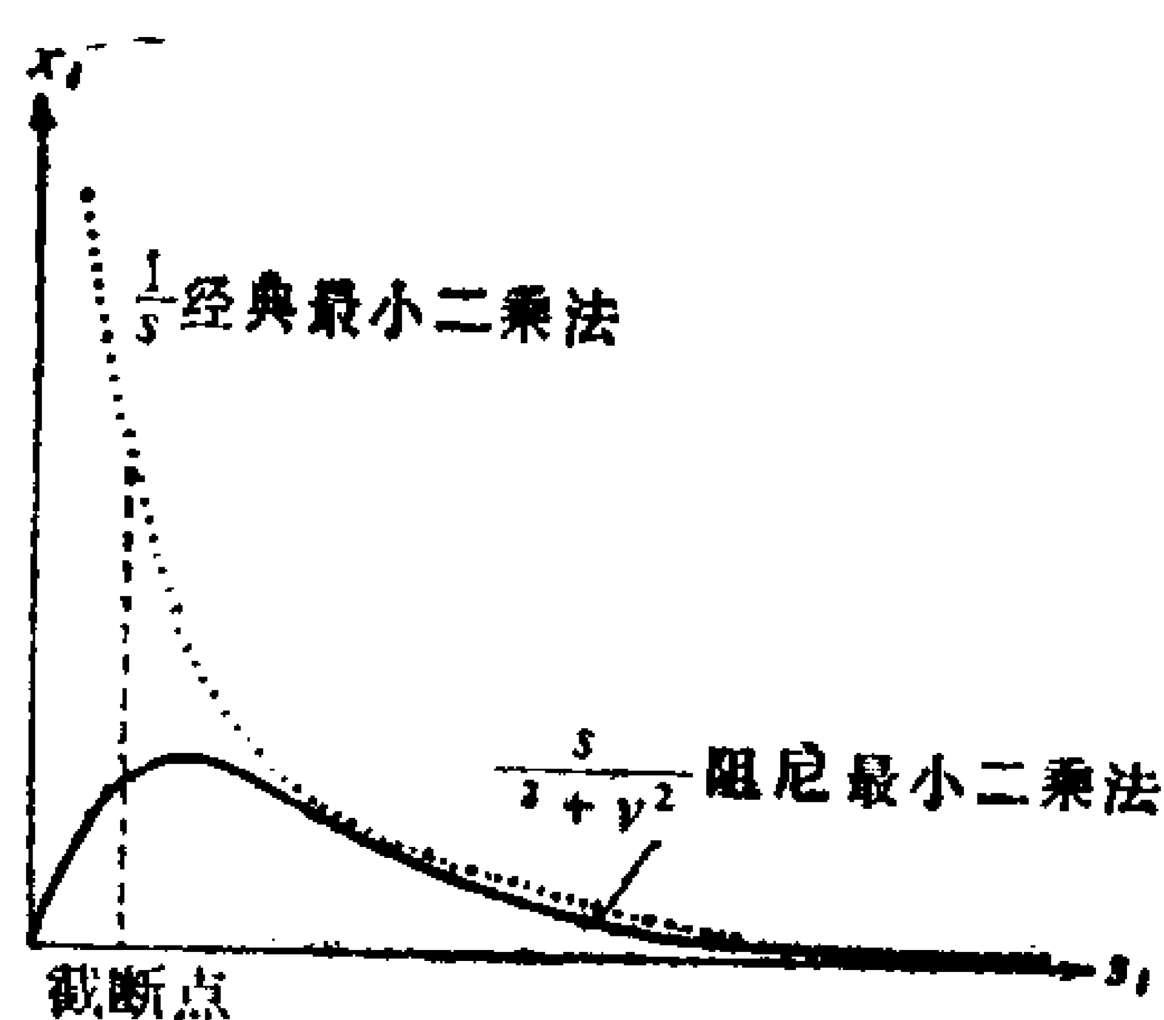



图 2-8-1

$S_i \to 0$ 时，解的某一分量$x$亦趋于零。这就保证了那些不提供信息的方向上的本征谱函数 $S_i/(S_i^2+\nu^2)$ 趋于零。

Wiggins(1972)曾经用突然截断本征值的办法来克服经典最小二乘法的发散问题(参见图2-8-1)。他考察本征值很小时的本征谱，建立截断本征谱，可以获得与阻尼最小二乘法类似的结果。

## §2-9　分辨矩阵、信息矩阵和协方差矩阵

用广义逆进行地球物理反演计算时，可以根据反演过程所用到的参数及矩阵，计算几个重要的补充信息，即分辨矩阵，信息矩阵及协方差矩阵。这些信息对评价反演结果有重要意义。以后，Crosson(1976) 又把这些计算推广到阻尼最小二乘法中去。

按照广义逆理论，对于任意 $\boldsymbol{A}_{m\times n}$ 矩阵构成的线性方程组

$$\boldsymbol{A}_{m\times n}\boldsymbol{x}_{n\times 1}=\boldsymbol{b}_{m\times 1}. \qquad (2\text{-}9.1)$$

总可以找到一个广义逆算子(按Moore-Penrose定义)

$$\boldsymbol{H}_{n\times m}=\boldsymbol{A}^{+}{}_{n\times m}. \qquad (2\text{-}9.2)$$

使得反演计算的解向量为

$$\hat{\boldsymbol{x}}=\boldsymbol{H}\boldsymbol{b}. \qquad (2\text{-}9.3)$$

如果广义逆矩阵$\boldsymbol{H}$满足下列三个准则，则它是一个令人满意的逆阵。或者说，能得到良好的反演结果。

准则1　$\boldsymbol{AH}=\boldsymbol{I}_m$，其中 $\boldsymbol{I}_m$ 为$m$阶单位矩阵。如果有 $\boldsymbol{AH}=\boldsymbol{I}_m$，则两端右乘 $\boldsymbol{b}$ 得

$$AHb=I_mb=b.\tag{2-9.4}$$

将(2-9.3)式代入(2-9.4)式，得到

$$A\hat{x}=b.\tag{2-9.5}$$

这表明由模型参量的反演计算值$\hat{x}$所计算的理论数据与实际观测数据$b$拟合得很好。因而$AH$与单位矩阵$I_m$的接近程度是理论计算值与观测值拟合好坏的一种衡量。

准则2　$HA=I_n$，其中$I_n$为$n$阶单位矩阵。如果有$HA=I_n$，则两端右乘$x$得

$$HAx=I_nx=x.\tag{2-9.6}$$

因$Ax=b$，故上式可以写为

$$Hb=x.$$

又据(2—9.3)式，上式可以写为

$$\hat{x}=x.\tag{2-9.7}$$

这表明，由反演计算得到的参数$\hat{x}$与“真解”$x$完全一致。由于这时只存在一个解，故$HA$与单位矩阵$I_n$的接近程度是解的唯一的一种衡量。

准则3　反演计算值$\hat{x}$的方差$\mathrm{Var}(\hat{x})$很小。对于统计上独立的观测数据，计算值$\hat{x}$的方差与观测值$b$的方差的关系为

$$\mathrm{Var}(\hat{x}_k)=\sum_{i=1}^{m}h_{ki}^2\mathrm{Var}(b_i).\tag{2-9.8}$$

式中$h_{ki}$为广义逆矩阵$H$的元素。准则3是衡量反演计算参数$\hat{x}$精度的标志。方差$\mathrm{Var}(\hat{x})$愈大，反演结果的误差愈大。

由这些准则，可以引出以下几个重要概念。

## （一）分辨矩阵

由式(2-9.3)及$b=Ax$可以得到

$$\hat{x}=Hb=HAx.\qquad(2\text{-}9.9)$$

定义$R=HA$为分辨(*Resolution*)矩阵。故上式可写为

$$\hat{x}=Rx.\qquad(2\text{-}9.10)$$

由此可见，分辨矩阵$R$($n$阶方阵)是把“真解”$x$映射到反演计算解$\hat{x}$的一个算子。$\hat{x}$的任一个元素$\hat{x}_k$，可以解释为分辨矩阵$R$的第$k$行元素与“真解”$x$的褶积。

换言之，$R$是联系“真解”$x$与计算解$\hat{x}$之间的滤波器或“窗口”函数。$R$矩阵的每一行相当于一个“窗口”。通过这个“窗口”可以将“真解”$x$变换为计算解$\hat{x}$。例如，反演计算解的第$k$个元素$\hat{x}_k$，就是通过第$k$个“窗口”（$R$矩阵的第$k$行）对“真解”$x$各元素的一次过滤结果，或者说是围绕$x_k$相邻各点“真值”的权平均值。图 2-9-1 是地球物理反演计算的几种分辨矩阵形式。显然，每一行主峰值愈尖锐，则分辨率愈高，反之愈低。图中 (a) 的分辨率低，(d)的分辨率较高。

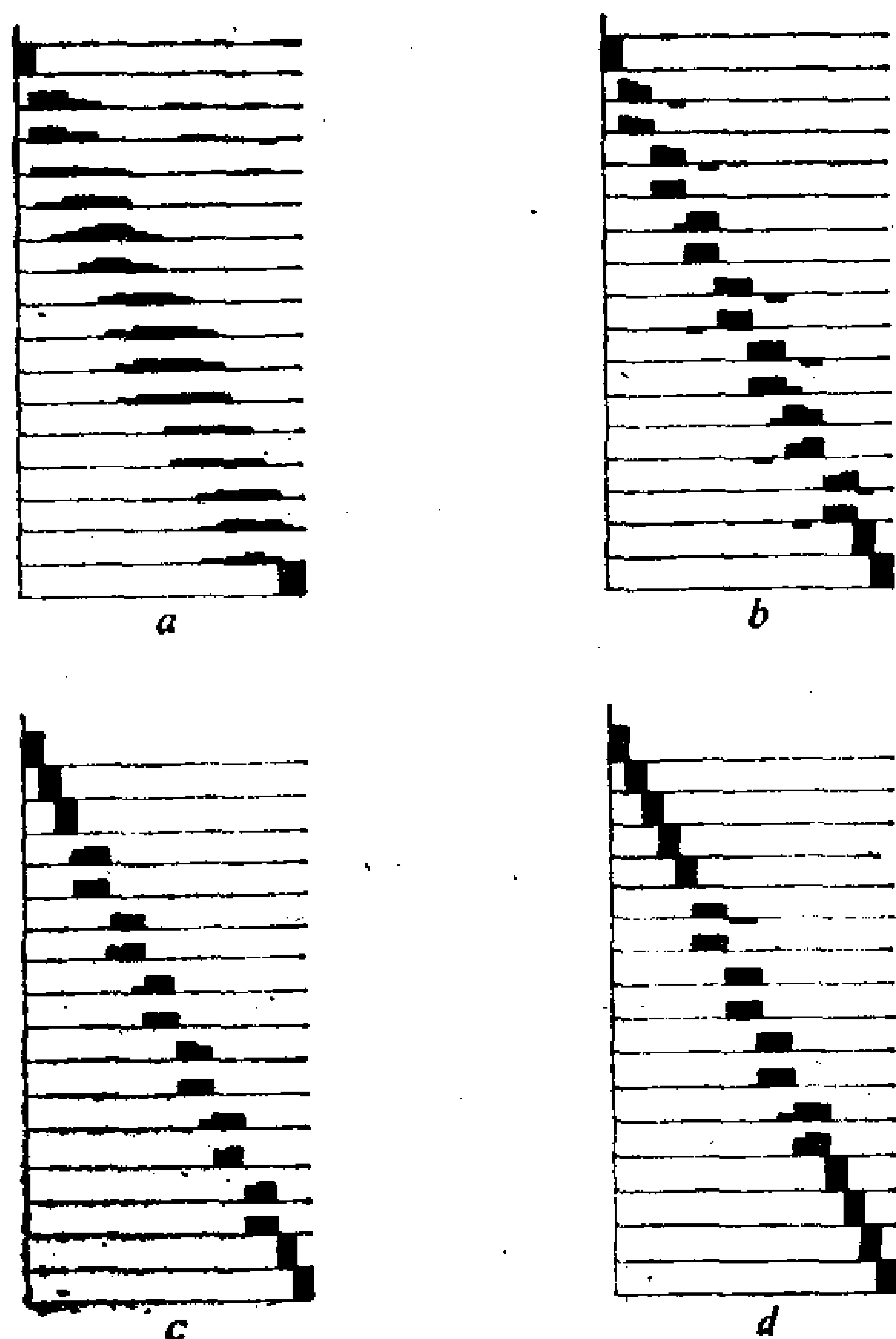



图 2-9-1

按照准则 2 可知，若分辨矩阵$R$是单位矩阵$I_n$，则$\hat{x}=x$。这表明反演计算解$\hat{x}$与“真解”$x$是一致的，即反演是唯一的。这种情况我们称为完全分辨。如果$R$近似于一个对角阵，则它是近似分辨的。若

$R$ 与单位矩阵偏离愈大，则分辨率愈差。这时，每一计算解的元素 $\hat{x}_k$ 实际上是“真解”$x_k$附近几个元素的加权和，$\hat{x}_k$与$x_k$相差较大。

下面来讨论分辨矩阵的计算方法。

对于一般情况，可对任意矩阵$A$作奇异值分解

$$A = USV^T.$$

相应的广义逆矩阵为

$$H = A^+ = VS^{-1}U^T.$$

故分辨矩阵

$$R = HA = VS^{-1}U^TUSV^T = VV^T. \qquad (2\text{-}9.11)$$

这表明分辨矩阵$R$仅与奇异值分解中正交矩阵$V$有关，说明$V$确定了参数反演的唯一性。故矩阵$V$中的向量称为参数特征向量。

对于地球物理反演中最经常出现的超定方程组，即对列满秩矩阵($r = n < m$)，由于其广义逆就是$A$的左逆

$$H = A^+ = (A^TA)^{-1}A^T, \qquad (2\text{-}9.12)$$

故

$$R = HA = (A^TA)^{-1}A^TA = I_n. \qquad (2\text{-}9.13)$$

这表明经典最小二乘法的分辨矩阵，即等于单位矩阵。故其反演是完全分辨的，或者反演具有唯一性。

若采用阻尼最小二乘法进行反演，对于$A$的奇异值分解式，分辨矩阵的计算公式为

$$R = V[(S^2 + \nu^2 I)^{-1}S^2]V^T. \qquad (2\text{-}9.14)$$

对于列满秩矩阵，分辨矩阵的计算公式为

$$R = (A^TA + \nu^2 I)^{-1}A^TA. \qquad (2\text{-}9.15)$$

以上二式中的$\nu$为阻尼因子。

## (二) 信息矩阵

对(2-9.3)式两端左乘$A$得

$$A\hat{x} = AHb. \tag{2-9.16}$$

令其左端$A\hat{x} = \hat{b}$，$\hat{b}$表示理论计算的“观测值”。换言之，它是由模型$\hat{x}$计算的理论值$\hat{b}$。

定义 $$AH = S \tag{2-9.17}$$

为信息矩阵或信息密度(*Information desity*)矩阵。于是(2-6.16)式可重写为

$$\hat{b} = Sb. \tag{2-9.18}$$

由上式不难看出，$S$是将实际观测值$b$变换为理论计算值$\hat{b}$的算子。$\hat{b}_k$是信息矩阵$S$的第$k$行元素与观测数据$b$的各元素的一个褶积。信息矩阵$S$是观测数据独立性的一种衡量。$S$与单位矩阵$I_m$愈接近，说明数据之间的独立性愈强，也就是计算值$\hat{b}_k$只与观测值$b_k$有关，而与$b_k$邻近的其它值无关。反之，若$S$偏离单位矩阵，则第$k$个计算的理论值$\hat{b}_k$不仅与观测值$b_k$有关，还和$b_k$邻近观测值有关，其独立性较差。从准则1 $AH = I_m$来看，信息矩阵$S$实际上表示了计算值$\hat{b}$与实测值$b$的符合程度，因此，它又是衡量理论计算值与实际观测值拟合好坏的一种量度。

信息矩阵$S$的对角元素$S_{ii}$的大小是权衡反演计算时第$i$个观测值$b_i$重要性的标志。因此，计算出信息矩阵$S$又可以确定那些观测值是重要的，那些是不重要的甚至是可以忽略的。这说明信息矩阵$S$还可以指导实际观测，帮助我们减少不重要的数据的观测，提高重要数据的观测精度。图2-9-2为信息矩阵的计算例子。

下面给出信息矩阵的计算方法。对于广义逆方法，一般是对任意$A$矩阵进行奇异值分解

$$A = USV^T, \quad H = VS^{-1}U^T.$$

得信息矩阵

$$S = AH$$
$$= USV^T VS^{-1} U^T$$
$$= UU^T. \quad (2\text{-}9.19)$$

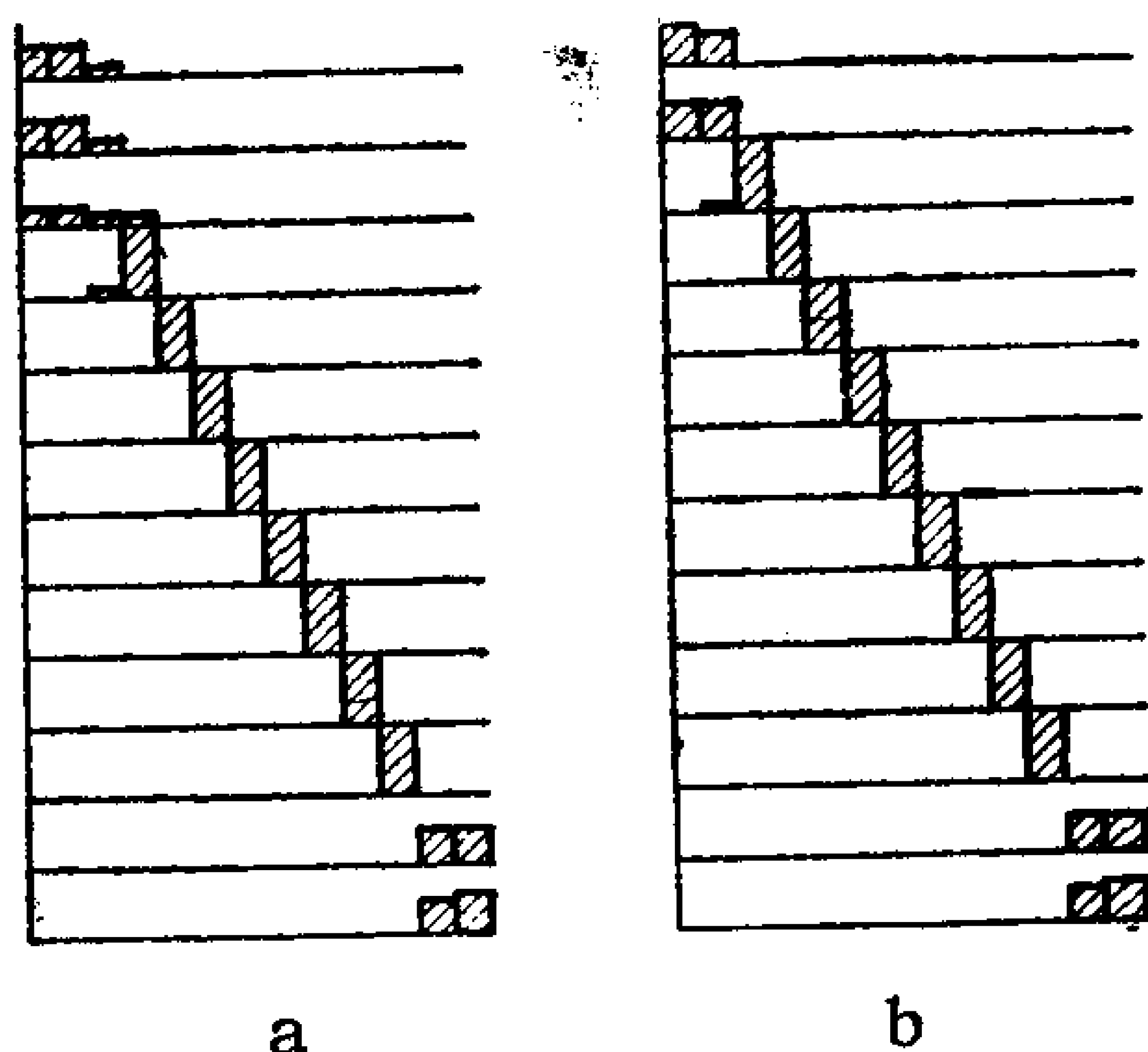


图 2-9-2

上式表明，信息矩阵$S$只与正交矩阵$U$有关。换言之，奇异值分解中的正交矩阵$U$决定了观测数据独立性的信息矩阵 $S$。所以，矩阵 $U$ 中的向量称为观测数据的特征向量。

对于在地球物理反演问题中常见的列满秩矩阵，其最小二乘解的信息矩阵为

$$S = A(A^T A)^{-1} A^T. \quad (2\text{-}9.20)$$

若用阻尼最小二乘法，由奇异值分解的计算公式得到的信息矩阵为

$$S = U[S^2(S^2 + \nu^2 I)^{-1}]U^T. \quad (2\text{-}9.21)$$

对于列满秩矩阵，阻尼最小二乘法的信息矩阵计算公式为

$$S = A(A^T A + \nu^2 I)^{-1} A^T. \quad (2\text{-}9.22)$$

## （三）协方差矩阵

由于反演计算的解答$\hat{x}$是多维随机变量，因此，我们不仅要考虑其中一个随机变量的方差，而且要考虑各随机机变量之间的相互关系，即协方差。例如，对于两个随机变量$\hat{x}_1$及$\hat{x}_2$，其协方差矩阵可以写为

$$C(\hat{x}) = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}. \quad (2\text{-}9.23)$$

其中各元素为

$$C_{ij}=E\{\hat{x}_i\hat{x}_j-E(\hat{x}_i)^2\},\quad i,\ j=1,\ 2. \qquad (2\text{-}9.24)$$

式中E表示求数学期望。

对于$n$个变量的情况，协方差矩阵的计算公式如下

$$C(\hat{\boldsymbol{x}})=E\{(\hat{\boldsymbol{x}}-E(\hat{\boldsymbol{x}}))(\hat{\boldsymbol{x}}-E(\hat{\boldsymbol{x}}))^T\}, \qquad (2\text{-}9.25)$$

其中$\hat{\boldsymbol{x}}=(\hat{x}_1,\ \hat{x}_2,\ \cdots,\ \hat{x}_n)^T$。

因此协方差矩阵中每一项元素可以写为

$$C_{ij}=E\{\hat{x}_i\hat{x}_j-E(\hat{x}_i)^2\},$$

$$i,\ j=1,\ 2,\ \cdots,\ n. \qquad (2\text{-}9.26)$$

写成矩阵为

$$\boldsymbol{C}=\begin{pmatrix} \mathrm{Var}(\hat{x}_1) & \mathrm{Cov}(\hat{x}_1,\hat{x}_2) & \mathrm{Cov}(\hat{x}_1,\hat{x}_3) & \cdots & \mathrm{Cov}(\hat{x}_1,\hat{x}_n) \\ \mathrm{Cov}(\hat{x}_2,\hat{x}_1) & \mathrm{Var}(\hat{x}_2) & \mathrm{Cov}(\hat{x}_2,\hat{x}_3) & \cdots & \mathrm{Cov}(\hat{x}_2,\hat{x}_n) \\ \mathrm{Cov}(\hat{x}_3,\hat{x}_1) & \mathrm{Cov}(\hat{x}_3,\hat{x}_2) & \mathrm{Var}(\hat{x}_3) & \cdots & \mathrm{Cov}(\hat{x}_3,\hat{x}_n) \\ \vdots & \vdots & \vdots & & \vdots \\ \mathrm{Cov}(\hat{x}_n,\hat{x}_1) & \mathrm{Cov}(\hat{x}_n,\hat{x}_2) & \mathrm{Cov}(\hat{x}_n,\hat{x}_3) & & \mathrm{Var}(\hat{x}_n) \end{pmatrix}.$$

$$(2\text{-}9.27)$$

其中主对角线多元素为方差

$$\mathrm{Var}(\hat{x}_i)=E\{\hat{x}_i^2-E(\hat{x}_i)^2\},$$

而其余各元素为协方差

$$\mathrm{Cov}(\hat{x}_i,\ \hat{x}_j)=E\{\hat{x}_i\hat{x}_j-E(\hat{x}_i)^2\}.$$

将(2-9.3)式代入(2—9.25)式，协方差矩阵可以写为

$$\begin{aligned} C(\hat{\boldsymbol{x}}) &= \mathrm{E}\{(\boldsymbol{Hb}-\mathrm{E}(\boldsymbol{Hb}))(\boldsymbol{Hb}-\mathrm{E}(\boldsymbol{Hb}))^T\} \\ &= \boldsymbol{H}\mathrm{E}\{(\boldsymbol{b}-\mathrm{E}(\boldsymbol{b}))(\boldsymbol{b}-\mathrm{E}(\boldsymbol{b}))^T\}\boldsymbol{H}^T \\ &= \boldsymbol{HC}(\boldsymbol{b})\boldsymbol{H}^T. \end{aligned} \qquad (2\text{-}9.28)$$

上式说明，反演计算参数$\hat{x}$的协方差矩阵$C(\hat{x})b$取决于观测值$b$的协方差矩阵$C(b)$。如果观测数据$b$中的分量$b_i$是彼此独立的，且是均值为零、方差为$\sigma^2$的随机变量。那么，(2-9.28)中的协方差矩阵$C(b)$中各交叉项为零，只存在主对角线各方差项$\sigma_b^2$，故上式可简化为

$$C(\hat{x})=\sigma_b^2 HH^T. \tag{2-9·29}$$

观测值方差$\sigma_v^2$可用下列无偏估计式计算，

$$\sigma_v^2=\frac{\Phi_{min}}{m-n}. \tag{2-9·30}$$

式中$\Phi_{min}$为（观测值与理论值）残差平方和的极小值（或目标函数极小值），可用下式计算，

$$\Phi_{min}=(b-A\hat{x})^T(b-A\hat{x}). \tag{2-9·31}$$

下面讨论协方差矩阵$C(\hat{x})$的计算公式。利用奇异值分解定理，广义逆$H=VS^{-1}U^T$，故其最小二乘解的协方差矩阵为

$$C(\hat{x})=\sigma_v^2 HH^T=\sigma_v^2 VS^{-1}U^TUS^{-1}V^T$$
$$=\sigma_v^2 VS^{-2}V^T. \tag{2-9·32}$$

对于阻尼最小二乘法，则协方差矩阵的计算公式为

$$C(\hat{x})=\sigma_v^2 V\{(S^2+\nu^2 I)^{-2}S^2\}V^T. \tag{2-9·33}$$

若为列满秩矩阵，则最小二乘解的协方差计算公式为

$$C(\hat{x})=\sigma_v^2(A^TA)^{-1}A^TA(A^TA)^{-1}=\sigma_v^2(A^TA)^{-1}, \tag{2-9·34}$$

而阻尼最小二乘法解答的协方差矩阵公式为

$$C(\hat{x})=\sigma_v^2(A^TA+\nu^2 I)^{-1}A^TA(A^TA+\nu^2 I)^{-1}. \tag{2-9·35}$$

计算出协方差矩阵$C(\hat{x})$后，其主对角线元素就是各相应反演参数的方差

$$Var(\hat{x}_k)=C_{kk},\ k=1,2,\cdots,n. \qquad (2\text{-}9\cdot36)$$

最后，我们考虑在反演的分辨率与模型参数可靠性之间取折衷(*Trade-off*)的问题。显然，我们希望由反演计算得到的参数的误差愈小愈好。但参数误差小会引起另一个问题，即反演结果是不稳定的(唯一性差)。这可由下列推导看出。

由(2-9.36)得知，某一反演参数的方差正是式(2-9.32)协方差矩阵中的对角元素

$$\sigma^2_{\hat{x}_k}=C_{kk}=\sigma^2_v\sum_{i=1}^{n}\left(\frac{v_{ki}}{S_i}\right)^2,k=1,2,\cdots,n. \qquad (2\text{-}9.37)$$

显然分子部份正好是分辨矩阵 $\boldsymbol{R}=\boldsymbol{V}\boldsymbol{V}^T$的元素。当分辨矩阵 $\boldsymbol{R}$ 为单位阵时，参数方差 $\sigma^2_{\hat{x}_k}$有极大值；反之，当分辨矩阵$\boldsymbol{R}$与单位阵相差较大时，$\sigma^2_{\hat{x}_k}$的值较小。这表明，对于给定的观测误差 $\sigma_b$，分辨率与反演参量的误差$\sigma_{\hat{x}_k}$是矛盾的。分辨率高，则反演参量误差较大；分辨率低，则反演参量误差较小。要使反演参量在精度和唯一性方面都得到照顾，必须在二者间采取一种折衷方案。

# 第三章　约束条件下反演问题的数值解法

在第一章里，我们已指出反演的不唯一性的根本原因在于信息不足及存在干扰。用无约束反演计算方法，往往收敛于局部极小值。因此，要取得较合理的反演结果，除了提高观测质量和计算精度，以增强信息分辨能力外。最重要的是将先验的或已知的地质和地球物理资料补充进来，作为反演计算的约束条件。

这一章，我们将讨论约束条件下反演问题的几种算法。第一种是直接处理约束条件的算法；第二种是解约束条件下的最小二乘问题；第三种是把约束最优化问题化为一系列无约束最优化问题求解。由于篇幅有限。我们只能介绍几种基本的和广泛使用的算法及计算机程序。更多的方法可参见线性或非线性规划有关专著。

## §3-1　直接处理约束条件的逐次归位算法

逐次归位法是一种直接处理约束条件的算法，简便易行。其基本想法是把求目标函数极小以及将参数归位到约束域。以上分为两个步骤进行，即先用无约束最优化方法求函数极小点，接着检查极小点是否在约束域内，如果出界则应进行归位计算。下面讨论具体算法。

从满足等式及不等式条件

$$
\begin{aligned}
&g_i(\boldsymbol{x})=0, \quad i=1,2,\cdots,p; \\
&h_j(\boldsymbol{x})\leqslant 0, \quad j=1,2,\cdots,q
\end{aligned}
\tag{3-1·1}
$$

的某一初始点 $x_0$ 出发，用某种最优化算法沿给定方向 $p$ 进行一维寻查，即求

$$\Phi(x_0+\lambda_0 p)=\min_{\lambda}\Phi(x_0+\lambda p). \qquad (3\text{-}1\cdot 2)$$

由此得到某一新点

$$x_1=x_0+\lambda_0 p. \qquad (3\text{-}1\cdot 3)$$

由于在作一维寻查时，仅保证目标函数$\Phi(x)$之值沿方向$p$达到最大下降，但并不能保证$x_0$满足约束条件，所以紧接着必须检验是否符合约束条件。

将$x_1$代入式(3-1.1)，若满足约束条件，则说明新点在约束可行域内。下一次寻查的起始点取$x_1$。

若$x_1$代入式(3-1.1)不满足约束条件，则说明新点$x_1$已超出约束可行域，应作归位计算。

归位计算的任务是使$x_1$重新回到约束条件的可行域内。也就是对向量$x_1$作一定的位移，使归位后的点

$$\tilde{x}_1=x_1+\Delta x_r, \qquad (3\text{-}1\cdot 4)$$

满足关系式(3-1.1)。$\Delta x_r$称为归位校正量。下面讨论怎样求归位校正量$\Delta x_r$。

归位校正量一方面是使归位后的点$\hat{x}_1$既在约束条件的可行域内，又使目标函数值进一步减小或至少不增大。因此，我们总是使归位点尽可能靠近寻查点，即归校正量总是取得尽量小($\min\|\Delta x_r\|$)，使满足约束条件

$$\begin{aligned} g_i(x_1+\Delta x_r)&=0, & i&=1,2,\cdots,p;\\ h_j(x_1+\Delta x_r)&\leqslant 0, & j&=1,2,\cdots,q. \end{aligned} \qquad (3\text{-}1\cdot 5)$$

若$g_i$及$h_j$为线性函数，则从(3-1.5)式中一次可求得校正量$\Delta x_r$。若$g_i$及$h_j$为非线性函数，则可在$x_1$点附近使$g_i$及$h_j$线性化，用迭代法逐次解出校正量$\Delta x_r$。然后取适合关系式(3-1.5)的校正量最小值$\min\|\Delta x_r\|$为实际校正量。

经过归位校正后的点$\tilde{x}_1$，就作为下一次迭代的起点，继续进行运算。在整个迭代过程中，反复交替使用求目标函数极小点和

归位两种运算，直到目标函数$\Phi(x)$达到预定的限差，或待求参数的修改量小到某一预定值，则反演计算完成。

直接处理约束条件的归位算法的程序设计简单，只须在任何一种无约束最优化或广义逆算法中加上归位校正子程序，就能达到目的。实际使用表明，其收敛速度及反演效果是良好的。具体应用及计算机程序可参看第八章面波反演部分。

## §3-2 约束条件下最小二乘问题解法

约束条件下任意线性方程组$Ax \cong b$的最小二乘解称为约束最小二乘问题 CLS(Constrained Least-Squares Problem)。根据约束条件的性质，又分为线性等式约束最小二乘问题 LSE (Linear Equality Constrained Least-Squares Problem) 和线性不等式约束最小二乘问题 LSI(Linear Inequality Constrained Least-Squares Problem)。

等式约束最小二乘问题LSE的建立：给定一个秩为$k_1$的$m_1 \times n$矩阵$C$及$m_1$维向量$d$，一个$m_2 \times n$矩阵$E$及$m_2$维向量$f$，对所有的$n$维向量$x$在满足约束条件

$$Cx = d \tag{3-2·1}$$

的情况下寻求

$$\|Ex - f\| \tag{3-2·2}$$

的极小佳值。

当且仅当(3-2.1)式独立时，LSE问题有解。我们将假定(3-2.1)式是相容的。对于实际问题来说，若满足$n > m_1 = k_1$的条件。我们就确信(3-2.1)式是独立的，且存在一个以上的解。

若LSE问题的解存在，则增广矩阵$\begin{bmatrix} C \\ E \end{bmatrix}$的秩为$n$时有唯一解，这是一个最小二乘最小范数解。下面介绍两种解法。

第一种是基于对增广矩阵$\begin{bmatrix}C\\E\end{bmatrix}$进行正交变换的零空间正交基方法。

若约束矩阵的正交分解为

$$C=HRK^T, \tag{3-2.3}$$

因为Rank$(C)=k_1$，且$k_1\leqslant n$，故正交矩阵$K$可以分为二块：

$$K=[\underbrace{K_1}_{k_1},\ \underbrace{K_2}_{n-k_1}]n. \tag{3-2·4}$$

于是解向量可以写为

$$x=C^+d+K_2y_2. \tag{3-2·5}$$

式中$y_2$为范围遍及全部$n-k_1$维向量空间的任意向量。

定理　设等式约束方程组$Cx=d$，则使范数$\|Ex-f\|$最小的解为

$$\hat{x}=C^+d+(EZ)^+(f-EC^+d). \tag{3-2·6}$$

式中　$Z=I_n-C^+C$。

式(3-2.6)也可以等价地表达为

$$\hat{x}=C^+d+K_2(EK_2)^+(f-EC^+d), \tag{3-2·7}$$

式中矩阵$K_2$由(3-2.4)定义。

当且仅当方程组$Cx=d$独立及增广矩阵$\begin{bmatrix}C\\E\end{bmatrix}$的秩为$n$时，此LSE问题的解向量$\hat{x}$是唯一的。

下面根据(3-2.7)式讨论计算方法。我们假定$C$矩阵为行满秩矩阵，即Rank$(C)=k_1=m_1$，这一条件对大多数实际问题是合理的。根据第二章叙述的QR分解定理，(3-2.3)式中的正交矩阵$H$可以视为单位阵。设$K$为$n\times n$正交矩阵，将$K$右乘$C$时，$C$将变换为下三角矩阵。于是得

$$\begin{bmatrix}C\\E\end{bmatrix}K=\begin{bmatrix}C\\E\end{bmatrix}[K_1\ \ K_2]$$

$$=\begin{pmatrix} CK_1 & CK_2 \\ EK_1 & EK_2 \end{pmatrix}\begin{matrix}\}m_1\\ \}m_2\end{matrix} = \begin{pmatrix} \widetilde{C}_1 & O \\ \widetilde{E}_1 & \widetilde{E}_2 \end{pmatrix}\begin{matrix}\}m_1\\ \}m_2\end{matrix} \tag{3-2·8}$$

$$\underbrace{}_{m_1}\quad\underbrace{}_{n-m_2}\qquad\qquad\underbrace{}_{m_1}\quad\underbrace{}_{n-m}$$

式中$\widetilde{C}$是非奇异$m_1 \times m_1$下三角阵。

解下列线性方程组

$$\widetilde{C}_1 \hat{y}_1 = d, \tag{3-2·9}$$

得$m_1$维向量$\hat{y}_1$，然后计算

$$\widetilde{f} = f - \widetilde{E}_1 \hat{y}_1 . \tag{3-2·10}$$

解下列最小二乘问题

$$\widetilde{E}_2 y_2 \cong \widetilde{f}, \tag{3-2·11}$$

得$(n-m_1)$维向量$\hat{y}_2$，最后，计算解向量

$$\hat{x} = K\begin{pmatrix}\hat{y}_1 \\ \hat{y}_2\end{pmatrix}. \tag{3-2·12}$$

下面举一个实例说明计算方法。设

$$C = [0.4087 \quad 0.1593],$$

$$E = \begin{bmatrix} 0.4302 & 0.3516 \\ 0.6246 & 0.3384 \end{bmatrix},$$

$$d = 0.1376,$$

$$f = \begin{bmatrix} 0.6593 \\ 0.9666 \end{bmatrix}.$$

首先根据QR分解方法，求出右乘矩阵$C$而使其变为三角阵的Householder正交矩阵$K$，也就是

$$K = \begin{bmatrix} -0.9317 & -0.3632 \\ -0.3632 & 0.9317 \end{bmatrix},$$

$K$ 可以通过专门的算法求出。利用式(3-2.8)计算

$$\begin{bmatrix} C \\ E \end{bmatrix} K = \begin{pmatrix} \widetilde{C}_1 & O \\ \widetilde{E}_1 & \widetilde{E}_2 \end{pmatrix} = \begin{pmatrix} -0.4386 & 0.0 \\ -0.5285 & 0.1714 \\ -0.7049 & 0.0885 \end{pmatrix},$$

即

$$\widetilde{C}_1 = [-0.4386],$$
$$\widetilde{E}_1 = \begin{bmatrix} -0.5285 \\ -0.7049 \end{bmatrix},$$
$$\widetilde{E}_2 = \begin{bmatrix} 0.1714 \\ 0.0885 \end{bmatrix}.$$

根据(3-2.9)式计算

$$\hat{y}_1 = -0.3137.$$

由(3-2.10)式计算

$$\widetilde{f} = \begin{bmatrix} 0.4935 \\ 0.7455 \end{bmatrix}.$$

由(3-2.11)式计算下列方程式的最小二乘解，

$$\begin{bmatrix} 0.1714 \\ 0.0885 \end{bmatrix} y_2 \cong \begin{bmatrix} 0.4935 \\ 0.7455 \end{bmatrix},$$

得 $\hat{y}_2 = 4.0472.$

于是由(3-2.12)式得到解答

$$\hat{x} = K \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \end{pmatrix} = \begin{bmatrix} -1.1775 \\ 3.8848 \end{bmatrix}.$$

我们来讨论等式约束最小二乘问题的另一种解法，即直接消元法。设增广矩阵$\begin{bmatrix} C \\ E \end{bmatrix}$可以分块为下列形式：

$$\begin{pmatrix} C \\ E \end{pmatrix} = \begin{pmatrix} C_1 & C_2 \\ \underbrace{E_1}_{m_1} & \underbrace{E_2}_{n-m_1} \end{pmatrix} \begin{matrix} \}m_1 \\ \}m_2 \end{matrix}, \tag{3-2·13}$$

其中 $C_1$ 中各列是线性独立的。且设

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} \begin{matrix} \}m_1 \\ \}n-m_1, \end{matrix} \tag{3-2·14}$$

则约束方程组$C\boldsymbol{x} = \boldsymbol{d}$可对$\boldsymbol{x}_1$求解，并得

$$\boldsymbol{x}_1 = C_1^{-1}(\boldsymbol{d} - C_2\boldsymbol{x}_2). \tag{3-2·15}$$

将$\boldsymbol{x}_1$的表达式代入(3-2.2)中，

$$\begin{aligned} \|E\boldsymbol{x} - \boldsymbol{f}\| &= \|E_1C_1^{-1}(\boldsymbol{d} - C_2\boldsymbol{x}_2) + E_2\boldsymbol{x}_2 - \boldsymbol{f}\| \\ &= \|(E_2 - E_1C_1^{-1}C_2)\boldsymbol{x}_2 - (\boldsymbol{f} - E_1C_1^{-1}\boldsymbol{d})\| \\ &\equiv \|\widetilde{E}_2\boldsymbol{x}_2 - \widetilde{\boldsymbol{f}}\|. \end{aligned} \tag{3-2·16}$$

令式(3-2.16)为极小，以解$\boldsymbol{x}_2$。由此可以得到下列消元解法。首先计算

$$\widetilde{E}_2 = E_2 - E_1C_1^{-1}C_2 \tag{3-2·17}$$

及

$$\widetilde{\boldsymbol{f}} = \boldsymbol{f} - E_1C_1^{-1}\boldsymbol{d}. \tag{3-2·18}$$

求下列线性方程组的最小二乘解得$\boldsymbol{x}_2$，

$$\widetilde{E}_2\boldsymbol{x}_2 \cong \widetilde{\boldsymbol{f}}. \tag{3-2·19}$$

最后计算

$$\boldsymbol{x}_1 = C_1^{-1}(\boldsymbol{d} - C_2\boldsymbol{x}_2). \tag{3-2·20}$$

具体的算法有许多。例如可以对 $C_1$ 进行 QR 分解，即用正交阵$Q_1$，左乘$C_1$ 得上三角阵

$$Q_1C_1 = \widetilde{C}_1,$$

故

$$C_1 = Q_1^{T}\widetilde{C}_1. \tag{3-2·21}$$

其中 $Q_1$ 为正交阵，$\widetilde{C}_1$ 为上三角阵。于是式(3-2·17)及(3-2·18)

可以写为

$$\widetilde{E}_2=E_2-(E_1\widetilde{C}_1^{-1})(Q_1C_2)\equiv E_2-\widetilde{E}_1\widetilde{C}_2, \quad (3\text{-}2\cdot 22)$$

$$\widetilde{f}=f-(E_1\widetilde{C}_1^{-1})(Q_1d)\equiv f-\widetilde{E}_1\widetilde{d}. \quad (3\text{-}2\cdot 23)$$

整个算法过程可归纳为如下步骤：

(1) 用Householder 变换使$C_1$三角化，并同时将它对$C_2$及$d$变换

$$Q_1[C_1 : C_2 : d]=[\widetilde{C}_1 : \widetilde{C}_2 : \widetilde{d}]. \quad (3\text{-}2\cdot 24)$$

(2) 计算 $m_2\times m_1$ 矩阵$\widetilde{E}_1$作为三角型方程组的解

$$\widetilde{E}_1\widetilde{C}_1=E_1. \quad (3\text{-}2\cdot 25)$$

(3) 计算

$$\widetilde{E}_2=E_2-\widetilde{E}_1\widetilde{C}_2 \quad (3\text{-}2\cdot 26)$$

及

$$\widetilde{f}=f-\widetilde{E}_1\widetilde{d}. \quad (3\text{-}2\cdot 27)$$

(4) 计算使$\widetilde{E}_2$三角化的Householder变换$Q_2$，并同时将它作用于$\widetilde{f}$，

$$Q_2[\widetilde{E}_2 : \widetilde{f}]=[\widehat{E}_2 : \widehat{f}]\}n-m_1. \quad (3\text{-}2\cdot 28)$$

(5) 最后通过下列三角型方程组计算解向量$\widehat{x}_1$及$\widehat{x}_2$

$$\begin{pmatrix}\widetilde{C}_1 & \widetilde{C}_2\\ O & \widetilde{E}_2\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix}=\begin{pmatrix}\widetilde{d}\\ \widehat{f}\end{pmatrix}. \quad (3\text{-}2\cdot 29)$$

对于前述的数值例子，利用(3-2•24)到(3-2•29)式可以计算出

$$\widetilde{E}_2=\begin{bmatrix}0.1839\\0.0949\end{bmatrix},$$

$$\widetilde{f}=\begin{bmatrix}0.5145\\0.7563\end{bmatrix},$$

$$\hat{x}=\begin{bmatrix}-1.1775\\3.8848\end{bmatrix}.$$

现在讨论不等式约束条件下最小二乘问题的解法。有关线性不等式约束条件下最小二乘问题 LSI 的一般提法如下：设 $E$ 为 $m_2\times n$ 矩阵，$f$ 为 $m_2$ 维向量，$G$ 为 $m\times n$ 矩阵，$h$ 为 $m$ 维向量，则线性不等式约束最小二乘问题LSI为：

在 $Gx\geqslant h$ 的条件下求 $\|Ex-f\|$ 极小。

LSI 有两类问题讨论得较详细。

第一类称非负约束最小二乘问题NNLS(Nonnegative Least Squares)，即：

在 $x\geqslant 0$ 条件下求 $\|Ex-f\|$ 极小。

第二类称为最短距离规划问题LDP (Least Distance Programming)，即：

在 $Gx\geqslant h$ 条件下求 $\|x\|$ 极小。

对于不等式约束最小二乘问题的一般算法已经在线性或非线性规划中详细讨论过。最先是Kuhn及Tucker(1951)提出这类问题的解法。他们引入了一个松弛或剩余参量 $r$，使不等式约束 $Gx>h$ 变为等式约束。$Gx-r=h(x,\ r>0)$。则 $\|Ex-f\|$ 最小问题可用目标函数 $\Phi(x)=\frac{1}{2}\|Ex-f\|^2$ 求极小来讨论。

LSI 问题的解可由下列定理来表达。

定理　一个 $n$ 维向量 $\hat{x}$ 为LSI 问题解的条件是，当且仅当存在一个 $m$ 维向量 $\hat{y}$ 和将整数1到 $m$ 分成两个子集 $A$（相等的）及 $S$（剩余的），满足关系式：

$$G^T\hat{y}=E^T(E\hat{x}-f),\tag{3-2·30}$$

$$\hat{r}_i=0,\ \hat{y}_i\geqslant 0,\quad i\in A;\tag{3-2·31}$$

$$\hat{r}_i>0,\ \hat{y}_i=0,\ i\in S.\tag{3-2·32}$$

其中

$$\hat{r}=G\hat{x}-h.\tag{3-2·33}$$

注意，上述定理中的向量

$$P=E^T(E\hat{x}-f)$$

是目标函数$\Phi(x)=\frac{1}{2}\|Ex-f\|^2$在$x=\hat{x}$处的梯度，而$-G^T$方向表示指向约束超平面外法线方向。

实际上，解LSI问题分为两步进行，第一步是将LSI归结为LDP问题，然后将LDP问题归结为NNLS问题。实际算法如下，

设LSI问题的$m_2\times n$矩阵$E$的秩为$k$，由奇异值分解定理

$$E=QRK^T$$

$$=[\underbrace{Q_1}_{k}\ \underbrace{Q_2}_{m_2-k}]\begin{bmatrix}R&0\\0&0\end{bmatrix}\begin{bmatrix}K_1^T\\K_2^T\end{bmatrix}\begin{matrix}\}k\\\}n-k\end{matrix}, \tag{3-2·34}$$

式中$Q$为$m_2\times m_2$正交矩阵，$K$为$n\times n$正交矩阵，$R$为$k\times k$非奇异对角阵。

引入对变量$x$的正交变换

$$x=K_1y, \tag{3-2·35}$$

则$\|Ex-f\|$极小问题可归结为求目标函数$\Phi(x)$的极小，即

$$\Phi(x)=\|f-Ex\|^2=\left\|\begin{bmatrix}Q_1^Tf\\Q_2^Tf\end{bmatrix}-\begin{bmatrix}Ry\\0\end{bmatrix}\right\|^2$$

$$=\|\tilde{f}_1-Ry\|^2+\|\tilde{f}_2\|^2. \tag{3-2·36}$$

其中 $$\tilde{f}_1=Q_1^Tf,\ \tilde{f}_2=Q_2^Tf. \tag{3-2·37}$$

若进一步令

$$z=Ry-\tilde{f}_1, \tag{3-2·38}$$

则(3-2.37)式可以写为

$$\Phi(x)=\|z\|^2+\|\tilde{f}_2\|^2. \tag{3-2·39}$$

于是求$Gx\geqslant h$条件下$\|f-Ex\|$极小的LSI问题，便归结为下列的LDP问题(除附加常数$\|\tilde{f}_2\|$外)：在$\tilde{G}z\geqslant\tilde{h}$的条件下求

$$\|z\| = \min, \tag{3-2·40}$$

其中
$$\widetilde{G} = GK_1R^{-1},$$

$$\widetilde{h} \geqslant h - GK_1R^{-1}\widetilde{f}_1.$$

第二步是将 LDP 问题转换为 NNLS 问题。这可以通过下述变化来达到。计算解NNLS 条件的向量 $u$，即计算满足条件$u>0$时的

$$\|Fu - g\| = \min. \tag{3-2·41}$$

其中
$$F = \begin{pmatrix} \widetilde{G}^T \\ \widetilde{h}^T \end{pmatrix},$$

g为 $n+1$ 维向量，除第 $n+1$ 个元素外，其余所有元素都等于零。

在 LDP 问题中，目标函数 $\frac{1}{2}\|z\|^2$ 的梯度向量简化为 $z$。Kuhn-Tucker 条件要求 $z$ 表达为$\widetilde{G}^T$行向量的非线性组合。于是表达为

$$z = \widetilde{G}^T u \|r\|^{-2}, \tag{3-2·42}$$

且剩余参量为

$$\|r\|^2 = r^T r = r^T \|Fu - g\|. \tag{3-2·43}$$

## §3-3 惩罚函数法

在解约束条件下地球物理反演问题时，另一类常用的有效方法是惩罚函数法。这种方法对于解不等式约束问题有较好的效果。它将约束问题转化为无约束最优化问题的序列，然后利用上一章讨论的无约束最优化方法求解。所以这种方法又称为序列无约束最优化方法 SUMT（Sequential Unconstrained Minimization Techniques）。

惩罚函数法的基本思想是将约束最优化问题转化为形如

$$\min_{x\in R^n} \left[ \Phi(x) + \mu_k P(x) \right] \tag{3-3·1}$$

的无约束最优化问题。其中$\{\mu_k\}$为一正常数单调递增序列，即$\mu_1 < \mu_2 < \cdots < \mu_k < \cdots$，$\lim\limits_{k\to\infty}\mu_k = \infty$。$\mu_k$从一般称为惩罚因子，而$\mu_k P(x)$则称为惩罚项。$P(x)$即是惩罚函数，一般应满足以下条件：

(1) $P(x)$为$R^n$上的连续函数；

(2) $P(x)>0$，当$x\overline{\in} S$时； (3-3·2)

(3) $P(x)=0$，当$x\in S$时。

为了计算上的方便，将显式约束(1-7·7)及隐式约束(1-7·8)以约束方程组$h_i(x)\leqslant 0$ $(i=1,2,\cdots,q)$代替，则惩罚函数不难根据(3-3·2)式条件而写为

$$P(x) = \sum_{i=1}^{q} (\max(0, h_i(x)))^2. \tag{3-3·3}$$

一般我们将(3-3.1)式称为增广目标函数求极小问题，将(3-3·3)式代入写为

$$\min_{x\in R^n} F(x,\mu) = \min_{x\in R^n}\left[\Phi(x) + \mu\sum_{i=1}^{q}(\max(0,h_i(x)))^2\right]. \tag{3-3·4}$$

可以证明存在下述定理：

(1) 若$x^*$是问题(1-7.10)的极小点，则$x^*$必为问题(3-3·4)的极小点。

(2) 由于$\mu_{k+1}>\mu_k$，并假定$F(x(\mu_k),\ \mu_k)$和$F(x(\mu_{k+1}),\ \mu_{k+1})$分别在$x^*(\mu_k)$和$x^*(\mu_{k+1})$达到它们各自在$R^n$中的极小值，则有下列不等式成立：

$$F(x^*(\mu_{k+1}),\ \mu_{k+1}) \geqslant F(x^*(\mu_k),\ \mu_k), \tag{3-3·5}$$

$$P(x^*(\mu_k)) \geqslant P(x^*(\mu_{k+1})), \tag{3-3·6}$$

$$\Phi(x^*(\mu_{k+1})) \geqslant \Phi(x^*(\mu_k)). \qquad (3\text{-}3.7)$$

(3) 设$\Phi(x)$，$h_i(x)(i=1,2,\cdots,q)$为$R^n$上的连续函数，则由惩罚法产生的序列$\{x^*(\mu_k)\}$的任一极限点是问题（1-7·10）的极小点。

以上几个定理说明了惩罚函数法的基本原理。由式(3-3·4)可以看出，对于不满足约束条件的点$x$($x\overline{\epsilon} S$)，惩罚项$\mu_k P(x) = \mu_k \sum_{i=1}^{q}(\max(0,h_i(x)))^2 > 0$，且随$\mu_k$增大而增大。它表示对于不满足约束条件的一种惩罚。当$x$点满足约束条件时($x\epsilon s$)，则有$\mu_k P(x)=0$，这时无论$\mu_k$取多大，增广目标函数$F(x(\mu_k),\mu_k)$与$\Phi(x)$取值相同。它表示由于满足约束条件而不受惩罚。(3-3·5)-(3-3·7)表明$F(x(\mu_k),\mu_k)$和$\Phi(x(\mu_k))$对于$\mu_k$是单调增加的，而P($x(\mu_k)$)对于$\mu_k$是单调减少的。因此当$\sum_{i=1}^{q}[\max(0,h_i(x(\mu_k)))]^2$充分小时，点$x(\mu_k)$将充分接近约束区域的边界。从某种意义上来看$x(\mu_k)$就可以看成满足约束条件的点。

我们举一个简单的例子来说明这一思想。设$h_1(x)=x-b$，$h_2(x)=a-x$，由约束条件限制的可行域S为闭区间$[a, b]$。图3-3-1表示出惩罚项$\mu P(x)$的图形。由图可以看出，$x(\mu_k)$随着$\mu_k$增加从外面逐渐趋向于可行域$S$。于是理想地当$\mu_k\to\infty$时，惩罚函数的解便收敛于约束问题的解。

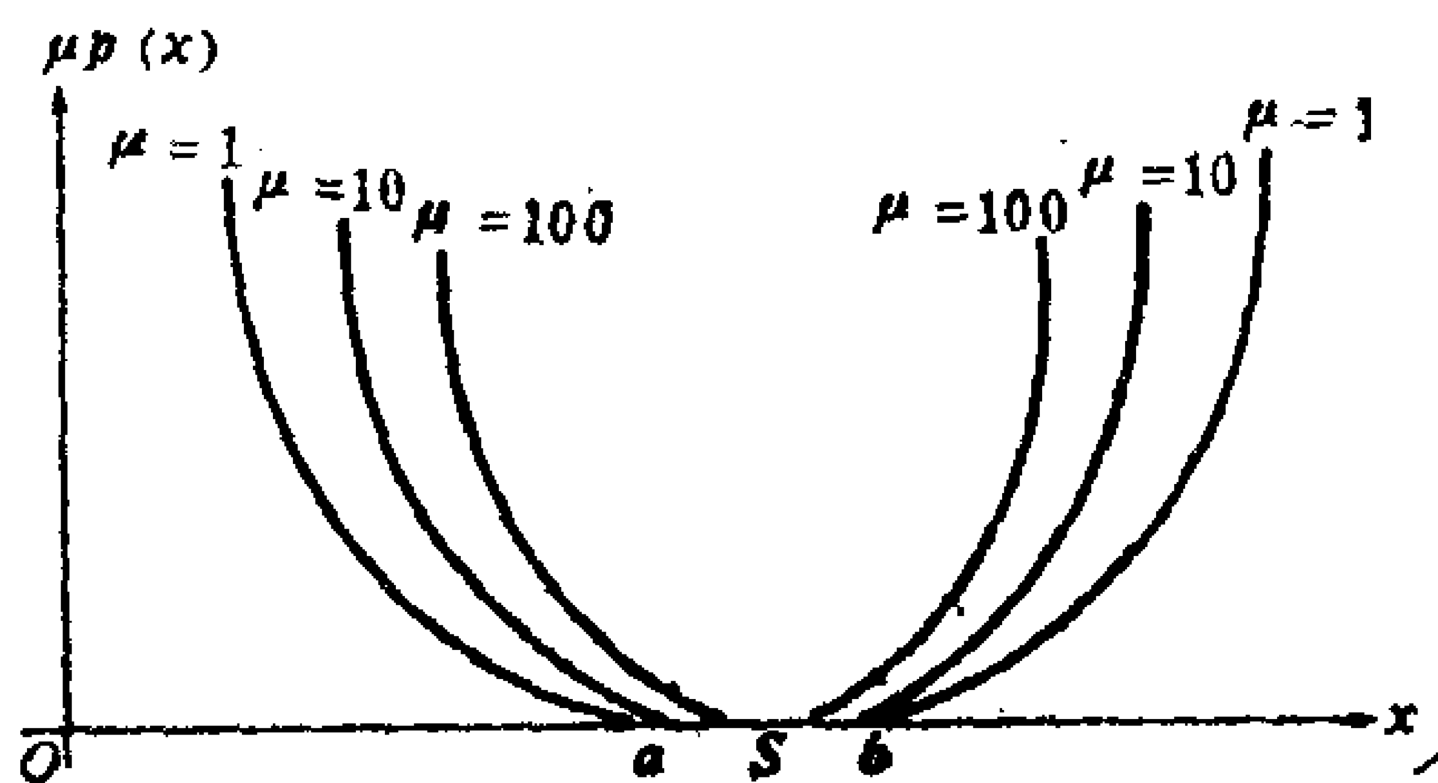


图 3-3-1

惩罚函数的一般计算步骤如下：

(1) 取$\mu_1>0$，给定参数初值$x_0\epsilon R^n$及精度要求$\varepsilon$。

(2) 假设利用某种无约束最优化方法求得

式(3-3.4)对应的近似极小点$x^*(\mu_k)\overline{\in}S$。

(3) 以$x^*(\mu_k)$为初始点，取$\mu_{k+1}>\mu_k$，继续用这种无约束最优化方法求解$\min\limits_{x\in R^n} F(x,\ \mu_{k+1})$得到相应的极小点$x^*(\mu_{k+1})$。

(4) 若对所有$i=1,2,\cdots,q$都满足

$$h_i(x^*(\mu_{k+1}))<\varepsilon,$$

则可取$x^*(\mu_{k+1})$为满足精度要求(3-3·4)式的极小点。否则取$\mu_{k+1}$，并以$x^*(\mu_{k+1})$为新的初始近似，转向第3步，继续迭代。

在实际的地球物理问题反演计算中，当利用惩罚函数法求增广目标函数极小时，我们采用收敛较快的阻尼最小二乘法。为了将惩罚项直接引入到阻尼最小二乘法的方程中去，对于线性不等式约束的情况作了以下的改进。

设存在线性不等式约束方程组

$$H(x)=Bx-C\leqslant O. \tag{3-3.8}$$

式中$B$为$q\times n$矩阵，而$c$为$q\times 1$向量。

为便于建立惩罚函数(3-3.3)式的矩阵形式，我们引入一个新的对角矩阵

$$w=\begin{pmatrix} w_1 & & & 0 \\ & w_2 & & \\ & & \ddots & \\ 0 & & & w_q \end{pmatrix}, \tag{3-3.9}$$

式中对角元素定义为

$$w_i=\begin{cases} 1 & h(x)>0 \ (\text{即 } x\overline{\in}S), \\ 0 & h_i(x)\leqslant 0 \ (\text{即 } x\in S). \end{cases}$$

于是惩罚函数可以写为

$$P(x)=\sum_{i=1}^{q}(w_ih_i(x))^2=(wBx-wc)^T(wBx-wc)$$

$$=(Bx-c)^Tw(Bx-c). \tag{3-3.10}$$

注意上式中有关系$w=w^T$及$w^Tw=w$。

则增广目标函数为

$$
\begin{aligned}
F(x,\ \mu) &= \Phi(x)+\mu P(x) \\
&= (y-f(x))^T(y-f(x))+\mu(Bx-c)^Tw(Bx-c).
\end{aligned}
\tag{3-3·11}
$$

对上式展开并取到一级近似值，其增量式可以写为

$$
\begin{aligned}
F(x_0+\delta,\ \mu) &= \Phi(x_0+\delta)+\mu P(x_0+\delta) \\
&= (y-f(x_0))^T(y-f(x_0))-2(y-f(x_0))^TA\delta \\
&\quad +\delta^TA^TA\delta+\mu\{(Bx_0-c)^Tw(Bx_0-c) \\
&\quad +2(B\delta)^Tw(Bx_0-c)+\delta^TB^TwB\delta\}.
\end{aligned}
\tag{3-3·12}
$$

式中$A$为 Jacobi 矩阵，$\{a_{ij}\}=\left.\frac{\partial f_i}{\partial x_j}\right|_{x=x_0}$。

按照增广目标函数极小化条件，令

$$
\begin{aligned}
\frac{\partial F(x_0+\delta,\ \mu)}{\partial\delta} &= -A^T(y-f(x_0))+A^TA\delta+\mu\{B^Tw(Bx_0-c) \\
&\quad +B^TwB\delta\}=0,
\end{aligned}
\tag{3-3·13}
$$

整理得

$$
(A^TA+\mu B^TwB)\delta=A^T(y-f(x_0))-\mu B^Tw(Bx_0-c).
\tag{3-3·14}
$$

于是，带惩罚项的阻尼最小二乘法便可写为

$$
((A^TA+\mu B^TwB)+\nu^2I)\delta=A^T(y-f(x_0))-\mu B^Tw(Bx_0-c),
$$

或者简化为

$$
(G+\nu^2I)\delta=g.
\tag{3-3·15}
$$

其中$G=A^TA+\mu B^TwB$，$g=A^T(y-f(x_0))-\mu B^Tw(Bx_0-c)$。

带惩罚项的阻尼最小二乘法框图如图3-3-2所示。

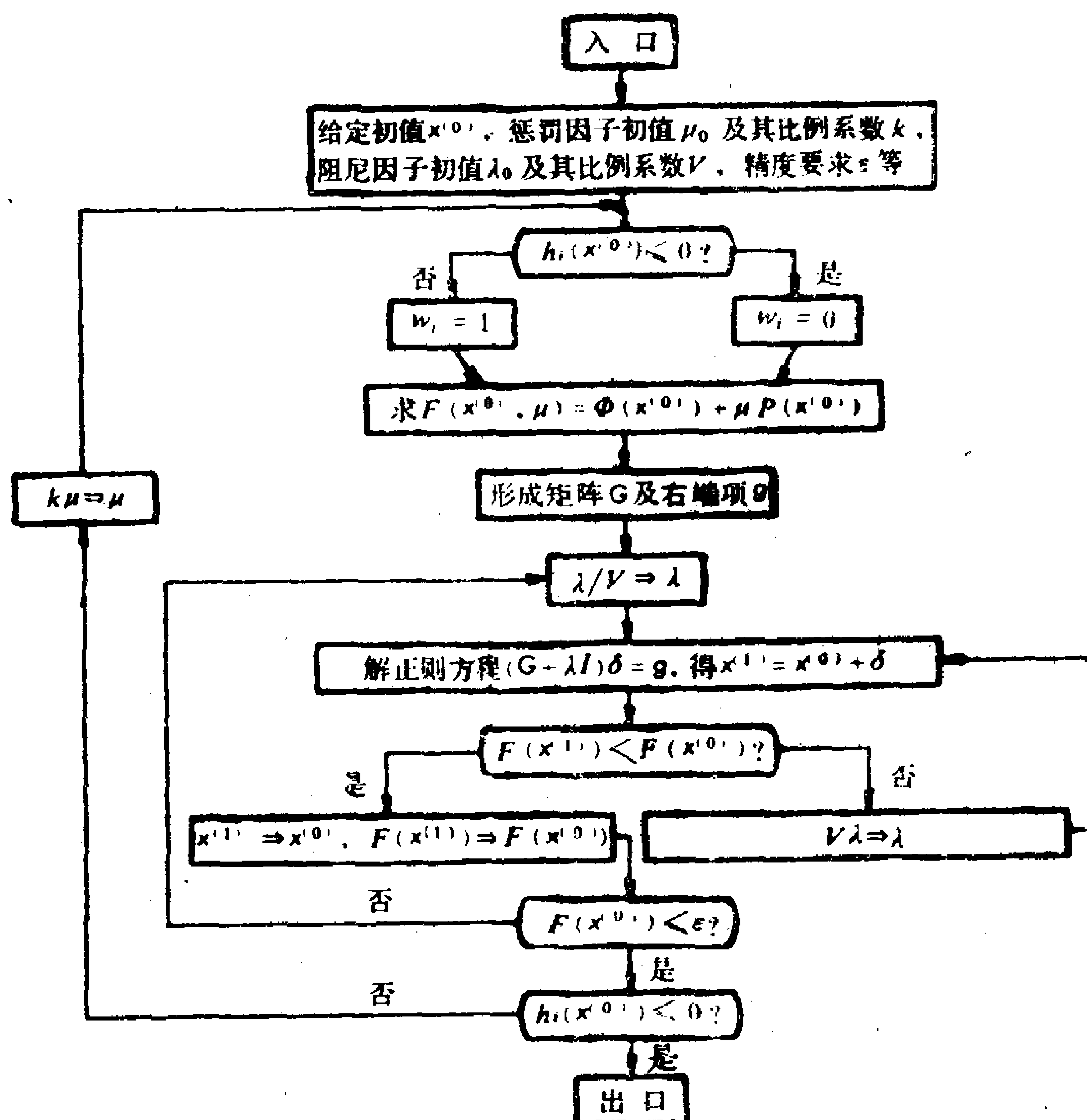



图 3-3-2

# §3-4 可变容限法

可变容限方法(The Flexible Tolerance Method)用于求

带有等式及不等式约束条件的目标函数极小问题：

$$\min \Phi(x), \qquad x \in R^n.$$

并满足约束条件

$$\left.\begin{aligned} &h_i(x)=0, && i=1,2,\cdots,m; \\ &g_i(x)\geqslant 0, && i=m+1,\cdots,p. \end{aligned}\right\} \tag{3-4·1}$$

此处$\Phi(x)$，$h_i(x)$，$g_i(x)$可以为线性或非线性函数。在其它许多约束最优化方法中，大部分的时间都花在满足严格的可行要求上。而可变容限的算法通过利用可行点，以及称为近似可行点的非可行点所提供的信息来改进目标函数值。当搜索极小的过程趋向于约束问题的解时，近似可行的条件逐渐变得严格，直到满足(3-4.1)式为止。

我们用一个更简单的形式来代替(3-4·1)式

$$\left.\begin{aligned} &\min \Phi(x),\ x\in R^n, \\ \text{满足约束}\quad &\phi^{(k)}-T(x)\geqslant 0. \end{aligned}\right\} \tag{3-4·2}$$

式中$\phi^{(k)}$是第$K$步搜索的可变容限标准的值，$T(x)$是由问题(3-4.1)所有等式或不等式约束所构成的一个正函数。本节所用的无约束最优化算法一般为单纯形法，所以$\phi^{(k)}$选择为$R^n$空间中活动多面体顶点正的非增函数：$\phi^{(k)}=\phi(x_1^{(k)},x_2^{(k)}\cdots,x_{r+1}^{(k)},x_{r+2}^{(k)})$，它的作用是作为整个搜索过程中对于违反约束的一个容许标准，同时也作为搜索结束时的一个准则，取

$$\phi^{(k)}=\min\{\phi^{(k-1)},\ \theta^{(k)}\}. \tag{3-4·3}$$

一般置 $\qquad \phi^{(0)}=2(m+1)t,$

$$\theta^{(k)}=\frac{m+1}{r+1}\sum_{i=1}^{r+1}\left\|x_i^{(k)}-x_{r+2}^{(k)}\right\|$$

$$=\frac{m+1}{r+1}\sum_{i=1}^{r+1}\left\{\sum_{j=1}^{n}\left(x_{i,j}^{(k)}-x_{r+2,j}^{(k)}\right)^2\right\}^{1/2}. \tag{3-4·4}$$

其中 $t$——初始多面体的大小因子；

$m$——等式约束的数目；

$n$——参变量数目；

$x_i^{(k)}$——$R^n$中多面体第$i$个顶点；

$r=n-m$ 自由度；

$x_{r+2}^{(k)}$ 单纯形法除去坏点后剩下点的质心坐标；

$k=0,1,\cdots$ 完成搜索次数指标。

从(3-4.4)式看到，$\theta^{(k)}$表示$R^n$中每个顶点$x_i^{(k)}$到多面体质心$x_{r+2}^{(k)}$的平均距离。虽然$\theta^{(k)}$在搜索中随单纯形的反射、扩张、收缩等可增可减，但$\phi^{(k)}=\min\{\phi^{(k-1)}, \theta^{(k)}\}$却为正的递减序列，即有

$$\phi^{(0)}\geqslant\phi^{(1)}\geqslant\cdots\geqslant\phi^{(k)}\geqslant 0. \qquad (3\text{-}4\cdot 5)$$

在单纯形算法中，随着搜索接近于解答时，多面体顶点与质心的平均距离收缩为零，由式(3-4.4)求出的$\theta^{(k)}$逐渐变小，随之$\phi^{(k)}$也减小。在极限情况下，$\phi^{(k)}$随着达到解答$x^*$而趋于零，

$$\lim_{x^{(k)}\to x^*}\phi^{(k)}=0. \qquad (3\text{-}4\cdot 6)$$

另外，我们把$T(x)$定义为等式与不等式约束的一个泛函，

$$T(x)=\left(\sum_{i=1}^{m}h_i^2(x)+\sum_{i=m+1}^{p}\delta_i g_i^2(x)\right)^{1/2}. \qquad (3\text{-}4\cdot 7)$$

其中

$$\delta_i=\begin{cases}0, & 当g_i(x)\geqslant 0;\\ 1, & 当g_i(x)<0.\end{cases} \qquad (3\text{-}4\cdot 8)$$

我们可以根据$T(x)$的值来区分可行与非可行点。设$\phi^{(k)}$为第$k$步搜索的$\phi$值，$x^{(k)}$为$R^n$中的任意向量，则有关系：

(1) 如$T(x^{(k)})=0$，$x^{(k)}$为可行点；

（2） 如$0 \leqslant T(x^{(k)}) \leqslant \phi^{(k)}$，$x^{(k)}$为近似可行点；

（3） 如$T(x^{(k)}) > \phi^{(k)}$，$x^{(k)}$为不可行点。

因此可以根据$T(x^{(k)})$值的大小及其趋于零的程度来判断$x^{(k)}$点周围可行域的情况。这是一个很有意义的改进，它有点类似于惩罚函数法中惩罚项的作用。

可变容限法的优点是当搜索趋向于解答时，约束条件是逐步达到的。这一算法与$\Phi(x)$的局部性质关系不大，因此可以避免一些不合需要的解。

实际计算时，可变容限法一般分为两步。第一步，对于给定的$\phi^{(k)}$值，找到使$T(x^{(k)}) = 0$或$T(x^{(k)}) < \phi^{(k)}$的$x^{(k)}$，第二步，以$x^{(k)}$为第k步搜索的初始点，寻找$\Phi(x^{(k+1)}) \leqslant \Phi(x^{(k)})$的点$x^{(k+1)}$，并使$\phi^{(k+1)} - T(x^{(k+1)}) \geqslant 0$。下面给出详细计算步骤：

（1） 用可变容限法求$\Phi(x)$极小，首先要给定初值$x^{(0)}$，$t$，$\phi^{(0)}$和$r$。$t$一般选为变量$x$所期望的变化范围的一个函数，通常对于$x_i$的上下界已知的情况，可用下面的公式估计$t$，

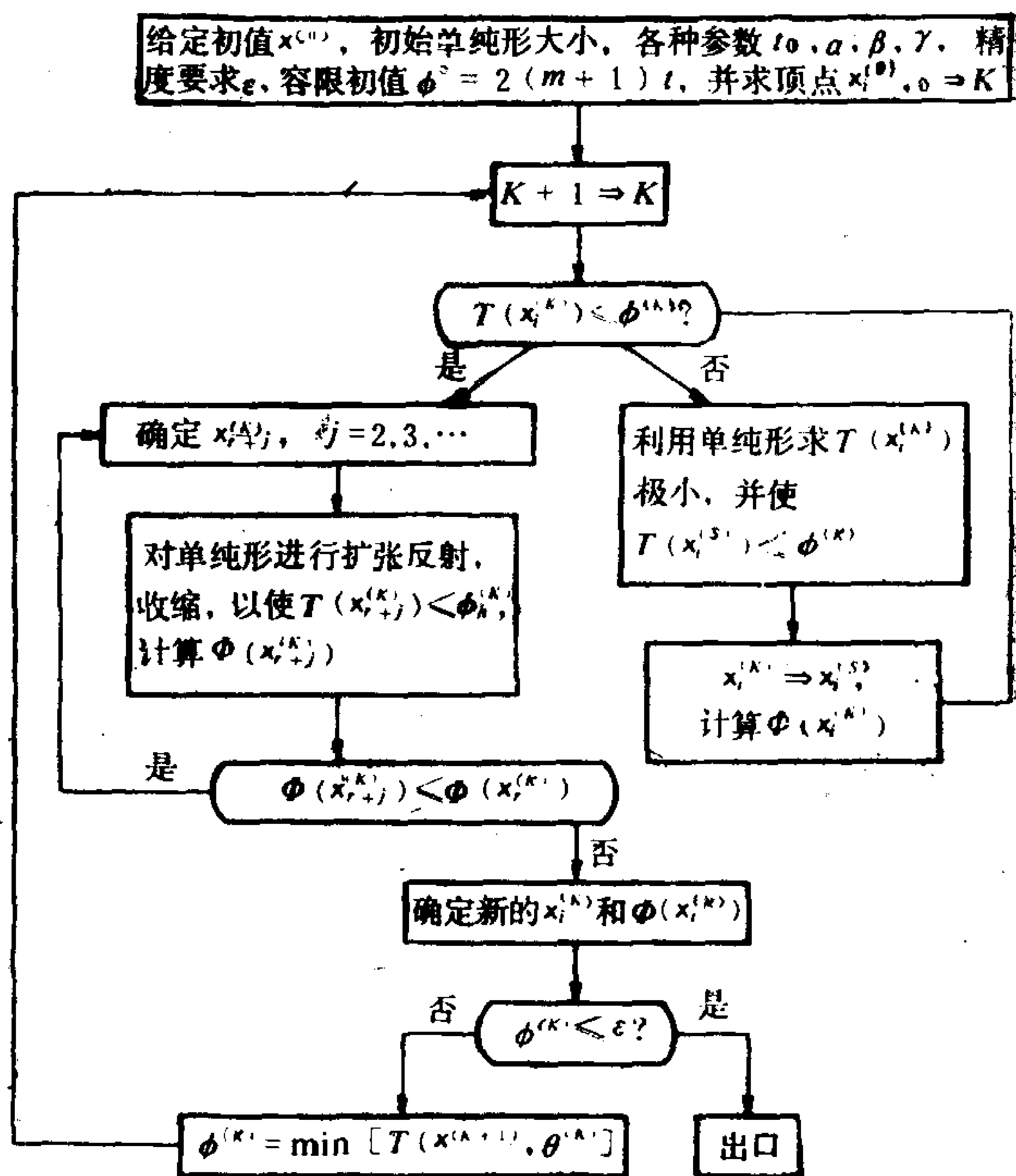
$$t = \min\left\{\frac{0.2}{n}\sum_{i=1}^{n}(u_i - l_i),\ (u_1 - l_1),\ \cdots,\ (u_n - l_n)\right\}. \tag{3-4·9}$$

式中$(u_i - l_i)$是第$i$个变量下上界的差。而$\phi^{(0)} = 2(m+1)t$，$r = n - m$。求出$r$，$t$，$\phi^{(0)}$就可以在$x^{(0)}$处计算$T(x)$的值。

（2） 设$x_i^{(0)}$为$R^n$中一个非可行点，$\phi^{(k)}$为第$k$步搜索的容限标准。将$t = 0.05\phi^{(k)}$作为初始多面体的大小，并用单纯形法确定$T(x)$极小化的$n+1$个顶点，

$$x_i^{(0)} = x_i^{(0)} + D_i,\quad i = 1,\ 2,\ \cdots,\ n+1. \tag{3-4·10}$$

$D_i$由单纯形法确定。设各顶点处$T(x)$值为$T(x_i^{(0)})$$(i = 1, 2, \cdots, n+1)$，且设单纯形法各系数为$\alpha = 1, \beta = 0.5$，$\gamma = 2$。用单纯形算法使$T(x)$极小化。并在$s$步后用$T(x_i^{(s)})$ $(i = 1,\ 2,\ \cdots,\ n+1)$



给定初值$x^{(0)}$，初始单纯形大小，各种参数$t_0$，$\alpha$，$\beta$，$\gamma$，精度要求$\varepsilon$，容限初值$\phi^0=2(m+1)t$，并求顶点$x_i^{(0)}$，$0\Rightarrow K$
$K+1\Rightarrow K$
$T(x_i^{(K)})\leqslant\phi^{(K)}$?
是
否
确定$x_{r+j}^{(K)}$，$j=2,3,\cdots$
利用单纯形求$T(x_i^{(K)})$极小，并使$T(x_i^{(S)})\leqslant\phi^{(K)}$
对单纯形进行扩张反射，收缩，以使$T(x_{r+j}^{(K)})<\phi_K^{(K)}$，计算$\Phi(x_{r+j}^{(K)})$
$x_i^{(K)}\Rightarrow x_i^{(S)}$，计算$\Phi(x_i^{(K)})$
是
$\Phi(x_{r+j}^{(K)})<\Phi(x_r^{(K)})$
否
确定新的$x_i^{(K)}$和$\Phi(x_i^{(K)})$
否
是
$\phi^{(K)}\leqslant\varepsilon$?
$\phi^{(K)}=\min[T(x^{(K+1)}),\theta^{(K)}]$
出口


图 5-4-1

中最小值$Tx(_0^{(s)})$与$\phi^{(k)}$相比较。

(3) 若$T(x_i^{(s)})\leqslant\phi^{(k)}$则确定了一个可行点或近似可行点。如果$Tx(_i^{(s)})>0$，就取$x_i^{(k)}=x_i^{(s)}$为近似可行点。并结束求$T(x)$的极小而开始求$\Phi(x)$的极小。如果$\Phi(x_i^{(s)})=0$且$m=0$，转到第6步。

(4) 若$T(x_i^{(s)})\geqslant\phi^{(k)}$，则计算

$$\lambda^{(s)}=\frac{1}{n+1}\left[\sum_{i=1}^{n+1}(T(x_i^{(s)})-T(x_{n+1}^{(s)}))^2\right]^{1/2}. \qquad (3\text{-}4.11)$$

式中$T(x_{n+2}^{(s)})$是在$T(x)$极小化第$s$步在多面体质心上的值。若$\lambda^{(s)}>10^{-7}$，则转向第2步继续进行$s+1$步$T(x)$的极小化。

(5) 若$\lambda^{(s)}\leqslant10^{-7}$，则可行多面体将变为一个点。这时再用单纯形法寻求极小将有很大困难。因而可沿每一坐标轴方向进行一维搜索。若某一方向新点$x_j^*$，有$T(x_j^*)\leqslant\phi^{(k)}$，则以$x_j^*$代替$x_i^{(k)}$，求$T(x)$最小过程结束。

若所有方向搜索后仍找不到一个可行或近似可行点，则转向第1步。如果连续三次重复1—5步都找不到，则以搜寻失败而结束。

(6) 用单纯形法求$\Phi(x)$极小。若$\Phi(x^{(k+1)})\leqslant\Phi(x^{(k)})$且$x^{(k+1)}$为可行或近似可行点，并有$\phi^{(k+1)}\leqslant\varepsilon$，则搜索结束，否则继续求$T(x^{(k+1)})$极小。

可变容限法计算框图如图3-4-1所示。

## §3-5 解约束最小二乘问题的计算程序和实例

本程序用奇异值分解算法解不等式约束最小二乘问题LSI。

其基本想法是把 LSI 问题转化为最短距离规划问题 LDP。然后再转为非负约束最小二乘问题NNLS。本程序有七个子程序及一个主程序，使用FORTRAN-Ⅳ语言编写。

## （一）标识符及子程序说明

### 1．最小二乘问题奇异值分解子程序SVDRS

对给定的$M\times N$矩阵$\boldsymbol{A}$及$M\times NB$矩阵$\boldsymbol{B}$，计算最小二乘问题$\boldsymbol{A}x\cong\boldsymbol{B}$的奇异值分解问题，

$$\boldsymbol{A}=\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T.$$

本子程序最后算出$\boldsymbol{S}$，$\boldsymbol{V}$及$\boldsymbol{G}=\boldsymbol{U}^T\boldsymbol{B}$。子程序中各形参说明如下：

A——二维数组，开始时存放$M\times N$矩阵A，计算结束时$N\times N$矩阵$\boldsymbol{V}$，$M\geqslant N$或$M<N$都允许。

B——可作一维或二维数组用，开始时存放$M\times NB$或$M\times 1$矩阵$\boldsymbol{B}$；计算结束时存放$M\times NB$或$M\times 1$矩阵$\boldsymbol{G}$。

S——中间工作单元，结束时前$N$个工作单元存放奇异值S(1)≥S(2)≥…≥S(N)≥0。

如果在调用子程序 QRBD时不能收敛，那么将打印信息“CONVERGENCE FAILURE IN QRBIDIAGONAL SVD ROUTINE”并停机。

### 2．双对角矩阵的奇异值分解子程序QRBD

对于$N\times N$双对角矩阵$B$进行奇异值分解

$$B=\widetilde{U}S\widetilde{V}^T.$$

本子程序执行 QR 分解算法。最后输出包括矩阵$B$的奇异值以及两个矩阵的积$V\widetilde{V}$及$\widetilde{U}^TC$。$V$及$C$分别为给定的$NRV\times N$和$N\times NC$矩阵。

$$\text{IPASS}=\begin{cases}1 & \text{QR 算法收敛，}\\ 2 & \text{作}10\times N\text{次QR 扫描后仍未收敛。}\end{cases}$$

D——开始存储$\boldsymbol{B}$的对角元素，结束时存储奇异值。

E——存放矩阵$B$的上对角元素。

N——矩阵$B$的阶次。

V——二维数组，$V$开始包括$NRV \times N$矩阵$V$，返回时将由$NRV \times N$的矩阵积$V\widetilde{V}$所置换。

C——二维数组$C$，开始时存放$N \times NC$矩阵$C$，输出时由积矩阵$\widetilde{U}^T C$代替。

**3．最短矩离规划子程序LDP**

计算下列约束最小二乘问题：在满足$Gx \geqslant h$条件下求$\|x\|$极小值。这里$G$为$M \times N$矩阵且$h$为$M$维向量。本程序调用NNLS子程序求解。其中：

G——LDP问题中的$M \times N$矩阵$G$。

H——M维向量$h$。

X——解向量$x$。

INDEX，W工作单元。

MODE 标志计算完成情况，1为成功；2表示维数不正确，如$N \leqslant 0$；3为子程序NNLS中最大迭代次数超过$3 \times M$；5表示不等式约束$Gx \geqslant h$不可解。

**4．非负约束最小二乘问题子程序NNLS**

计算满足$x \geqslant 0$时$Ax \cong b$的解。其中

A——$M \times N$矩阵$A$。

B——$M$维向量$b$。

X——$x \geqslant 0$的解向量。

RNORM 计算结束时残差向量的范数$\|b - Ax\|$。

W——对偶向量$w = A^T(b - Ax)$。

Z——工作单元。

INDFX——工作单元。

MODE——计算完成情况及结束原因，1表示成功；2表示维数错误，即$M \leqslant 0$或$N \leqslant 0$出现；3表示迭代次数超过$3 \times N$。

**5．构造和应用Householder 变换的子程序H12**

给定$M$维向量$v$及整数$l_p$及$l_1$，本子程序计算$M$维向量$u$及一个数$S$，构成$M\times M$ Householder 对称正交矩阵$Q=I+(uu^T)/(S_{l_p})$使满足$QV=W$，其中$w_i=v_i$（对$i<l_1$）$|w_{l_p}|=\left(v_{l_p}^2+\sum_{i=l_1}^{m}v_i^2\right)^{1/2}$，$w_i=v_i$（对$l_p<i<l_1$），以及$w_i=0$（对$l_1\leqslant i\leqslant M$）。

将$Q$矩阵乘以给定的$M$维向量。其中

MODE——使用时置1执行算法$H1$，置2时执行算法$H2$。

LP，L1，M——当满足$1\leqslant LP<L1\leqslant M$时，它们构成量$l_P$，$l_1$和$M$。如果不满足不等式时，子程序不执行计算并返回。

U——Householder 变换中的向量$u$。

C——中间工作单元。

**6．构造及应用旋转矩阵子程序G1及G2**

对给定的两个数$x_1$及$x_2$，子程序G1计算Givens正交旋转矩阵使

$$G\begin{bmatrix}x_1\\x_2\end{bmatrix}=\begin{bmatrix}\cos\theta & \sin\theta\\-\sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix}=\begin{bmatrix}(x_1^2+x_2^2)^{1/2}\\0\end{bmatrix}.$$

子程序G2执行矩阵——向量乘积计算

$$G\begin{pmatrix}Z_1\\Z_2\end{pmatrix}=\begin{pmatrix}\tilde{Z}_1\\\tilde{Z}_2\end{pmatrix}.$$

**7．函数子程序DIFF**

用以计算二个实数的差。

## （二）程序使用及实例

在主程序中对$G$，$E$和$h$，$f$直接赋值。

本程序给出一个计算实例。对于下述不等式约束最小二乘问题LSI：

在$Gx \geqslant h$条件下求$\|Ex - f\|$极小。

其中

$$E=\begin{pmatrix}0.25 & 1\\ 0.50 & 1\\ 0.50 & 1\\ 0.80 & 1\end{pmatrix},\quad f=\begin{pmatrix}0.5\\ 0.6\\ 0.7\\ 1.2\end{pmatrix},$$

$$G=\begin{pmatrix}1 & 0\\ 0 & 1\\ -1 & -1\end{pmatrix},\quad h=\begin{pmatrix}0\\ 0\\ -1\end{pmatrix}.$$

为将 LSI 问题转化为 LDP 问题，我们对矩阵 $E$ 计算正交分解，这里 $E$ 的QR分解和奇异值分解都要用到。奇异值分解式为

$$E=U_{4\times 4}\begin{bmatrix}S_{2\times 2}\\ O_{2\times 2}\end{bmatrix}V^T_{2\times 2},$$

$$S=\begin{bmatrix}2.255 & 0.0\\ 0.0 & 0.346\end{bmatrix},$$

$$V=\begin{bmatrix}-0.467 & 0.884\\ -0.884 & -0.467\end{bmatrix},$$

$$\begin{matrix}2\{\\ 2\{\end{matrix}\begin{pmatrix}\tilde{f}_1\\ \tilde{f}_2\end{pmatrix}=U^Tf=\begin{pmatrix}-1.536\\ 0.384\\ -0.054\\ 0.174\end{pmatrix}.$$

引入变量

$$z=SV^Tx-\tilde{f}_1.$$

于是问题便转为求下列LDP问题：

满足$\tilde{G}z \geqslant \tilde{h}$条件下求$\|z\|$的极小。

其中

$$\tilde{G} = GVS^{-1} = \begin{pmatrix} -0.207 & 2.558 \\ -0.392 & -1.351 \\ 0.599 & -1.206 \end{pmatrix},$$

以及

$$\tilde{h} = h - \tilde{G}\tilde{f}_1 = \begin{pmatrix} -1.300 \\ -0.084 \\ 0.384 \end{pmatrix}.$$

解答 $\hat{z}$ 为满足上述约束条件的最小欧氏范数。用子程序LDP计算得

$$\hat{z} = \begin{bmatrix} 0.127 \\ -0.255 \end{bmatrix}.$$

于是最后可计算解答

$$\hat{x} = VS^{-1}(\hat{\ } + \tilde{f}_1) = \begin{bmatrix} 0.621 \\ 0.379 \end{bmatrix}.$$

解 $\hat{x}$ 的残差向量为

$$\hat{r} = f - E\hat{x} = \begin{pmatrix} -0.034 \\ -0.089 \\ 0.011 \\ 0.324 \end{pmatrix}.$$

残差范数为

$$\|\hat{r}\| = 0.338.$$

## (三) FORTRAN-Ⅳ语言源程序

```
C          MAIN    PROGRAM
      DIMENSION E(4, 2), F(4), G(3, 2), H(3), G2(3, 2),
     * H2(3), X(2), Z(2), W(4)
      DIMENSION WLDP(21), S(6), T(4)
      INTEGER INDEX(3)
      WRITE(6, 110)
      MDE=4
      MDGH=3
```

```
      ME = 4
      MG = 3
      N = 2
C         DEFINE THE LEAST SQUARES  AND CONSTRA-
C         INT MATRICES
      T(1) = 0.25
      T(2) = 0.5
      T(3) = 0.5
      T(4) = 0.8
      W(1) = 0.5
      W(2) = 0.6
      W(3) = 0.7
      W(4) = 1.2
      DO    10    I = 1, ME
      E(I, 1) = T(I)
      E(I, 2) = 1.0
   10 F(I) = W(I)
      G(1, 1) = 1.0
      G(1, 2) = 0.0
      G(2, 1) = 0.0
      G(2, 2) = 1.0
      G(3, 1) = -1.0
      G(3, 2) = -1.0
      H(1) = 0.0
      H(2) = 0.0
      H(3) = -1.0
C         COMPUTE THE SINGULAR VALUE DECOMPOSI-
C         TION OF THE MATRIXE.
      CALL SVDRS (E, MDE, ME, N, F, 1, 1, S)
      DO 12 I = 1, N
      DO 11 J = 1, N
   11 E(I, J) = -E(I, J)
```

```
12  F(I)=-F(I)
    WRITE  (6, 120)  ((E(I, J), J=1, N), I=1, N)
    WRITE (6, 130)  F, (S(J), J=1, N)
C          DEFINE THE CONSTRAINT MATRIX FOR THE
C          Z-C-OORDINATE SYSTEM
    DO 30 I=1, MG
    DO 30 J=1, N
    SM=0.0
    DO 20 L=1, N
20  SM=SM+G(I, L)*E(L, J)
30  G2(I, J)=SM/S(J)
C          DEFINE CONSTRAINT RT SIDE FOR THE Z-CO-
C          ORDINATE SYSTEM
    DO 50 I=1, MG
    SM=0.0
    DO 40 J=1, N
40  SM=SM+G2(I, J)*F(J)
50  H2(I)=H(I)-SM
    WRITE(6, 140)  ((G2(I, J), J=1, N), I=1, MG)
    WRITE(6, 150)H2
C          SOLVE THE CONSTRAINED PROBLEM IN Z-C-
C          OORDINATES.
    CALL LDP  (G2, MDGH, MG, N, H2, Z, ZNORM,WL-
      DP, INDEX, MODE)
    WRITE (6, 200)  MODE, ZNORM
    WRITE(6, 160)Z
C          TRANSFORM BACK FROM Z-COORDINATES TO
C          X-COORDINATES
    DO 60 J=1, N
60  Z(J)=(Z(J)+F(J))/S(J)
    DO 80 I=1, N
    SM=0.0
```

```
      DO 70 J=1, N
   70 SM=SM+E(I, J)*Z(J)
   80 X(I)=SM
      RES=ZNORM**2
      NP1=N+1
      DO 90 I=NP1, ME
   90 RES=RES+F(I)**2
      RES=SQRT(RES)
C          COMPUTE THE RESIDUALS
      DO 100 I=1, ME
  100 F(I)=W(I)-X(I)*T(I)-X(2)
      WRITE(6, 170) (X(J), J=1, N)
      WRITE(6, 180) (I, F(I), I=1, ME)
      WRITE(6, 190) RES
      STOP
  110 FORMAT (36HO EXAMPLE OF CONSTRAINED CUR-
      VE FITTING ,26H USING THE SUBROUTINE LDP.,
      /43H0RELATED INTERMEDIATE QUANTITIES ARE
      1GRVE 2N.)
  120 FORMAT (10H0V=, 2F10.5/(10X, 2F10.5))
  130 FORMAT (10H0F TILDA=, 4F10.5/10H0S=, 2F10.5)
  140 FORMAT (10H0G TILDA=, 2F10.5/(10X, 2F10.5))
  150 FORMAT (10H0H TILDA=, 3F10.5)
  160 FORMAT (10H0Z=, 2F10.5)
  170 FORMAT (52H0THE COEFICIENTS OF THE FITTED
         1LINE F(T)=X(1)*T+X(2), 12H
         1ARE X(1)=, F10.5, 14H,      AND X(2)=, F10.5)
  180 FORMAT (30H0THE CONSECUTIVE RESIDUALS
         ARE/11X, 4(110, F10.5))
  190 FORMAT (23H0THE RESIDUALS NORM IS, F10.5)
  200 FORMAT (18H0MODE (FROM LOP)=, I3, 2X, 7HZN-
      1ORM=, F10.5)
```

```
      END
      SUBROUTINE SVDRS (A, MDA, MM, NN, B, MDB,
         NB, S)
      DIMENSION A(MDA, NN), B(MDB, NB), S(NN, 3)
      ZERO=0
      ONE=1
C     BEGIN SPECIAL FOR ZERO ROWS AND COLS.PA-
C     CK THE NONZERO COLS TO THE LEFT
      N=NN
      IF(N.LE.0.OR.MM.LE.0)RETURN
      J=N
   10 CONTINUE
      DO 20 I=1, MM
      IF(A(I, J)) 50, 20, 50
   20 CONTINUE
      IF(J.EQ.N) GO TO 40
      DO 30 I=1, MM
   30 A(I, J)=A(I, N)
   40 CONTINUE
      A(1, N)=J
      N=N-1
   50 CONTINUE
      J=J-1
      IF(J.GE.1) GO TO 10
C     IF N=0 THEN A IS ENTIRELY ZERO AND SVD C-
        OMPUTION CAN BE SKIPPED
      NS=0
      IF(N.EQ.0) GO TO 240
C     PACK NONZERO ROWS TO THE TOP QUIT PACKI-
C     NG IF FIND N NONZERO ROWS
      I=1
      M=MM
```

```
60    IF(I.GT.N.OR.I.GE.M)  GO  TO 150
      IF(A(I, I))  90, 70, 90
70    DO 80 J=1, N
      IF(A(I, J))  90, 80, 90
80    CONTINUE
      GO TO 100
90    I=I+1
      GO TO  60
      ROW I IS ZERO EXCHANGE ROWS I AND M
100   IF(NB.LE.0)  GO TO  115
      DO 110 J=1, NB
      T=B(I, J)
      B(I, J)=B(M, J)
110   B(M, J)=T
115   DO 120 J=1, N
120   A(I, J)=A(M.J)
      IF(M.GT.N)  GO TO  140
      DO 130 J=1, N
130   A(M, J)=ZERO
140   CONTINUE
C        EXCHANGE IS FINISHED
      M=M-1
      GO  TO 60
150   CONTINUE
      L=MIN0(M, N)
C      THE FOLLOWING LOOP REDUCES A TO UPPER B-
C      IDIAGONAL AND
C      ALSO APPLIES THE PREMULTIPLYING TRANSFO-
C        RMATIONS TO B
      DO 170 J=1, L
      IF(J.GE M) GO TO 160
      CALL H12 (1, J, J+1, M, A(1, J), 1, T, A(1, J+1),
```

```
     1, MDA, N-J)
     CALL H12(2, J, J+1, M, A(1, J), 1, T, B, 1, MDB,
        NB)
 160 IF(J.GE.N-1) GO TO 170
     CALL H12(1, J+1, J+2, N, A(J, 1), MDA, S(J, 3),
        A(J+1, 1), MDA, 1, M-J)
 170 CONTINUE
C       COPY THE BIDIAGONAL MATRIX INTO THE ARR-
C          AY S( ) FOR QRBD
     IF(N.EQ.1) GO TO 190
     DO 180 J=2, N
     S(J, 1)=A(J, J)
 180 S(J, 2)=A(J-1, J)
 190 S(1, 1)=A(1, 1)
     NS=N
     IF(M.GE.N) GO TO 200
     NS=M+1
     S(NS, 1)=ZERO
     S(NS, 2)=A(M, M+1)
 200 CONTINUE
C       CONSTRUCT THE EXPLICIT N BY N PRODUCT M-
C       ATRIX
C       W=Q1*Q2*. . . *QL*I IN THE ARRAY A( )
     DO 230 K=1, N
     I=N+1-K
     IF(I.GE.N-1) GO TO 210
     CALL H12(2, I+1, I+2, N, A(I, 1), MDA, S(I, 3),
     A(1, I+1), 1, MDA, N-I)
 210 DO 220 J=1, N
 220 A(I, J)=ZERO
 230 A(I, I)=ONE
C       COMPUTE THE SVD OF THE BIDIAGONAL MATRIX
```

```
      CALL QRBD(IPASS, S(1, 1), S(1, 2), NS, A, MDA,
         N, B, MDB NB)
      GO TO (240, 310), IPASS
  240 CONTINUE
      IF(NS.GE.N)  GO  TO 260
      NSP1 = NS + 1
      DO 250 J = NSP1, N
  250 S(J, 1) = ZERO
  260 CONTINUE
      IF(N.EQ.NN)  RETURN
      NP1 = N + 1
C         MOVE RECORD OF PERMUTATIONS AND STOR
             ZEROS
      DO 280 J = NP1, NN
      S(J, 1) = A(1, J)
      DO270 I = 1, N
  270 A(I, J) = ZERO
  280 CONTINUE
C     PERMUTE ROWS AND SET ZERO SINGULAR VAL
C         UES
      DO 300K = NP1, NN
      I = S(K, 1)
      S(K, 1) = ZERO
      DO 290 J = 1, NN
      A(K, J) = A(I, J)
  290 A(I, J) = ZERO
      A(I, K) = ONE
  300 CONTINUE
C               END. . SPECIAL FOR ZERO ROWS AND
C         COLUMNS
      RETURN
  310 WRITE(6, 320)
```

```
      STOP
  320 FORMAT(50H0 CONVERGENCE FAILURE IN QR BI-
         DIAGONAL SVD ROUTINE)
      END
      SUBROUTINE  QRBD(IPASS, Q, E, NN, V, MDV, N-
        RV C, MDC, NCC)
C     QR ALGORITHM FOR        SINGULAR VALUE OF
C     A BIDIAGONAL  MATRIX.
      LOGICAL WNTV, HAVERS, FAIL
      DIMENSION Q(NN),E(NN),V(MDV,NN),C(MDC,NCC)
      ZERO=0
      ONE=1
      TWO=2
      N=NN
      IPASS=1
      IF(N, LE.0)RETURN
      N10=10*N
      WNTV=NRV.GT.0
      HAVERS=NCC.GT.0
      FAIL=.FALSE.
      NQRS=0
      E(1)=ZERO
      DNORM=ZERO
      DO 10J=1, N
   10 DNORM=AMAX1(ABS(Q(J))+ABS(E(J)), DNORM)
      DO 200 KK=1, N
      K=N+1-KK
C       TEST FOR SPLITTING OR RANK DEFICIENCIES.
C           FIRST MAKE
C             TEST FOR LAST    DIAGONAL TERM, Q(K),
C            BEING SMALL.
   20 IF(K .EQ. 1)  GO TO 50
```

```
      IF(DIFF(DNORM+Q(K), DNORM)) . 50, 25, 50
C        SINCE Q(K) IS SMALL WE WILL MAKE A SP-
C        ECIAL PASS TO TRAN.E(K)  TO 0
   25 CS=ZERO
      SN= -ONE
      DO 40 II=2, K
      I=K+1-II
      F= - SN * E(I+1)
      E(I+1)=CS * E(I+1)
      CALL G1 (Q(I), F, CS, SN, Q(I))
C        TRANSFORMATION CONSTRUCTED TO ZERO PO-
C          SITION (I, K).
      IF(.NOT.WNTV)  GO TO 40
      DO 30 J=1, NRV
   30 CALL G2(CS, SN, V(J, I), V(J, K))
C          ACCUMULATE RT.TRANSFORMATIONS IN V
   40 CONTINUE
C       THE MATRIX IS NOW BIDIAGONAL, AND OF LO-
           WER ORDER SINCE E(K).EQ.0
   50 DO 60 LL=1, K
      L=K+1-LL
      IF(DIFF(DNORM+E(L), DNORM))  55, 100, 55
   55 IF(DIFF(DNORM+Q(L-1), DNORM))  60, 70, 60
   60 CONTINUE
C       THIS LOOP CAN, T COMPLETE SINCE E(1)=ZERO
      GO TO 100
C          CANCELLATION OF E(L), L.GT.1.
   70 CS=ZERO
      SN= -ONE
      DO 90 I=L, K
      F= - SN * (I)
      E(I) =CS * E(I)
```

```
      IF(DIFF(DNORM+F, DNORM)) 75, 100, 75
   75 CALL G1(Q(I), F, CS, SN, Q(I))
      IF(.NOT.HAVERS) GO TO 90
      DO 80 J=1, NCC
   80 CALL G2(CS, SN, C(I, J), C(L-1, J))
   90 CONTINUE
C     TEST FOR CONVERGENCE
  100 Z=Q(K)
      IF(L.EQ.K) GO TO 170
C     SHIFT FROM BOTTOM 2 BY2 MINOR OF B**(T)*
      X=Q(L)
      Y=Q(K-1)
      G=E(K-1)
      H=E(K)
      F=((Y-Z)*(Y+Z)+(G-H)*(G+H)/(TWO*H*Y)
      G=SQRT(ONE+F**2)
      IF(F.LT.ZERO) GO TO 110
      T=F+G
      GO TO 120
  110 T=F-G
  120 F=((X-Z)*(X+Z)+H*(Y/T-H))/X
C     NEXT QR SWEEP
      CS=ONE
      SN=ONE
      LP1=L+1
      DO 160 I=LP1, K
      G=E(I)
      Y=Q(I)
      H=SN*G
      G=CS*G
      CALL G1(F, H, CS, SN, E(I-1))
      F=X*CS+G*SN
```

```
      G=-X*SN  +G*CS
      H=Y*SN
      Y=Y*CS
      IF(.NOT.WNTV) GO TO 140
C      ACCUMULATE ROTATIONS (FROM THE RIGHT) IN
C      'V'.
      DO 130 J=1, NRV
  130 CALL G2(CS, SN, V(J, I-1), V(J, I))
  140 CALL G1(F, H, CS, SN, Q(I-1))
      F=CS*G+SN*Y
      X=-SN*G+CS*Y
      IF(.NOT.HAVERS) GO TO 160
      DO 150 J=1, NCC
  150 CALL G2(CS, SN, C(I-1, J), C(I, J))
C      APPLY ROTATIONS FROM THE LEFT TO RIGHT S
C      IDES IN'C'.
  160 CONTINUE
      E(L)=ZERO
      E(K)=F
      Q(K)=X
      NQRS=NQRS+1
      IF(NQRS.LE.N10) GO TO 20
C      RETURN TO'TEST FOR SPLITTING'
      FAIL=.TRUE.
C      CUTOFF FOR CONVERGENCE FAILURE'NQRS' WIL
C        LBE 2*NUSUALLY.
  170 IF(Z.GE.ZERO) GO TO 190
      Q(K)=-Z
      IF(.NOT.WNTV) GO TO 190
      DO 180 J=1, NRV
  180 V(J, K)=-V(J, K)
  190 CONTINUE
```

```
C        CONVERCENCE.Q(K) IS MADE NONNEGATIVE··
  200  CONTINUE
       IF(N.EQ.1)  RETURN
       DO 210 I=2, N
       IF(Q(I).GT.Q(I-1)) GO TO 220
  210  CONTINUE
       IF (FAIL) IPASS=2
       RETURN
            EVERY SINGULAR VALUE IS IN ORDER
  220  DO 270 I=2, N
       T=Q(I-1)
       K=I-1
       DO 230 J=I, N
       IF(T.GE.Q(J)) GOTO 230
       T=G(J)
       K=J
  230  CONTINUE
       IF(K.EQ.I-1) GO TO 270
       Q(K)=Q(I-1)
       Q(I-1)=T
       IF(.NOT.HAVERS) GOTO 250
       DO 240 J=1, NCC
       T=C(I-1, J)
       C(I-1, J)=C(K, J)
  240  C(K, J)=T
  250  IF(.NOT.WNTV) GO TO 270
       DO 260 J=1, NRV
       T=V(J, I-1)
       V(J, I-1)=V(J, K)
  260  V(J, K)=T
  270  CONTINUE
C           END OF ORDERING ALGORITHM
```

```
      IF(FAIL)  IPASS=2
      RETURN
      END
      SUBROUTINE LDP(G, MDG, M, N, H, X, XNORM,
     *   W, INDEX, MODE)
C          * * *LEAST DISTANCE  PROGRAMMING * * *
      INTEGER    INDEX (M)
      DIMENSION G(MDG, N), H(M), X(N), W(1)
      ZERO=0.0
      ONE=1.0
      IF(N.LE.0) GO TO  120
      DO 10 J=1, N
   10 X(J)=ZERO
      XNORM=ZERO
      IF(M.LE.0)  GO TO  110
      IW=0
      DO 30 J=1, M
      DO 20 I=1, N
      IW=IW+1
   20 W(IW)=G(J, I)
      IW=IW+1
   30 W(IW)=H(J)
      IE=IW+1
C        STORE N ZEROS FOLLOWED  BY A ONE INTO F
      DO 40 I=1, N
      IW=IW+1
   40 W(IW)=ZERO
      W(IW+1)=ONE
      NP1=N+1
      IZ=IW+2
      IY=IZ+NP1
      IWDUAL=IY+M
```

```
      CALL NNLS  (W, NP1, NP1, M, W(IE), W(IY), RN
        ORM,W(IWDUAL), W(IZ), INDEX, 1MODE)
C          USE THE FOLLOWING RETURN IF UNSUCC-
C        ESSFULIN NNLS
      IF(MODE.NE.1)  RETURN
      IF(RNORM)  130, 130, 50
   50 FAC=ONE
      IW=IY-1
      DO 60 I=1, M
      IW=IW+1
C        HERE WE ARE USING THE SOLUTION VECTOR Y
   60 FAC=FAC-H(I)*W(IW)
      IF(DIFF(ONE+FAC, ONE))  130, 130, 70
   70 FAC=ONE/FAC
      DO 90 J=1, N
      IW=IY-1
      DO 80 I=1, M
      IW=IW+1
   80 X(J)=X(J)+G(I, J)*W(IW)
   90 X(J)=X(J)*FAC
      DO 100 J=1, N
  100 XNORM=XNORM+X(J)**2
      XNORM=SQRT(XNORM)
C                             SUCCESSFUL RETURN
  110 MODE=1
      RETURN
C                 ERROR    RETURN.    N.LE.0
  120 MODE=2
      RETURN
C                    RETURNING WITH CONSTRAINTS NOT
        COMPATIBLE
  130 MODE=4
```

```
      RETURN
      END
      SUBROUTINE NNLS(A, MDA, M, N, B, X, RNORM,
         W, ZZ, INDEX, MODE)
C      * * * NONNEGATIVE  LEAST  SQUARES * * *
      DIMENSION A(MDA, N), B(M), X(N), W(N), ZZ(M)
      INTEGER INDEX(N)
      ZERO = 0.0
      ONE = 1.0
      TWO = 2.0
      FACTOR = 0.01
      MODE = 1
      IF(M.GT.0.AND.N.GT.0)  GO  TO 10
      MODE = 2
      RETURN
   10 ITER = 0
      ITMAX = 3 * N
C      INITIALIZE THE ARRYS INDEX(  )    AND X(  )
      DO 20 I = 1, N
      X(I) = ZERO
   20 INDEX(I) = I
      IZ2 = N
      IZ1 = 1
      NSETP = 0
      NPP1 = 1
C           * * * MAIN LOOP BEGINS HERE * * *
   30 CONTINUE
C          QUIT IF ALL COEFFICIENTS ARE ALREADY IN
C            THE
C          SOLUTION OR IF M COLS OF A HAVE BEEN
             TRIANGULARIZED
      IF(IZ1.GT.IZ2.OR.NSETP.GE.M)  GOTO 350
```

```
C         COMPUTE COMPONENTS OF THE DUAL (NEGAT-
C            IVE GRADINT)VECTOR W(  )
      DO 50 IZ=IZ1, IZ2
      J=INDEX(IZ)
      SM=ZERO
      DO 40 L=NPP1, M
   40 SM=SM+A(L, J)*B(L)
   50 W(J)=SM
C         FIND    LARGEST  POSITIVE  W(J)
   60 WMAX=ZERO
      DO  70 IZ=IZ1, IZ2
      J=INDEX(IZ)
      IF(W(J).LE.WMAX)  GO  TO 70
      WMAX=W(J)
      IZMAX=IZ
   70 CONTINUE
C         IF WMAX.LE.0, GOTO TERMINATION.THIS IND-
          ICATES SATISFACTION
C            OF THE KUHN-TUCKER   CONDITIONS
      IF(WMAX) 350, 350, 80
   80 IZ=IZMAX
      J=INDEX(IZ)
C         THE SIGN OF W(J) IS OK FOR J TO BE MOVED
C            TO SET P.BEGIN THE TRANS
C         -FORMATION AND CHECK NEW DIAGONL ELEM-
C            ENT TO AVOID NEAR LINEAR DEPENDENCE
      ASAVE=A(NPP1, J)
      CALL H12 (1, NPP1, NPP1+1, M, A(1, J), 1, UP, D-
        UMMY, 1, 1, 0)
      UNORM=ZERO
      IF(NSETP.EQ.0)  GO  TO 100
      DO 90 L=1, NSETP
```

```
90    UNORM=UNORM+A(L, J) * * 2
100   UNORM=SQRT(UNORM)
      IF(DIFF(UNORM÷ABS(A(NPP1, J)) * FACTOR, UNO-
         RM))130, 130, 110
110   DO 120 L=1, M
120   ZZ(L)=B(L)
      CALLH12(2, NPP1, NPP1+1, M, A(1, J), 1, UP, ZZ,
         1, 1, 1)
      ZTEST=ZZ(NPP1)/A(NPP1, J)
C           SEE IF ZTEST IS POSITIVE
      IF(ZTEST)130, 130, 140
130   A(NPP1, J)=ASAVE
      W(J)=ZERO
      GO TO 60
140   DO 150 L=1, M
150   B(L)=ZZ(L)
      INDEX(IZ)=INDEX(IZ1)
      INDEX(IZ1)=J
      IZ1=IZ1+1
      NSETP=NPP1
      NPP1=NPP1+1
      IF(IZ1.GT.IZ2)  GO  TO 170
      DO 160 JZ=IZ1, IZ2
      JJ=INDEX(JZ)
160   CALL H12(2, NSETP, NPP1, M, A(1, J), 1, UP, A
         (1, JJ), 1, MDA, 1)
170   CONTINUE
      IF(NSETP.EQ.M)  GO  TO 190
       DO 180 L=NPP1, M
180   A(L, J)=ZERO
190   CONTINUE
       W(J)=ZERO
```

```
C         SOLVE THE TRIANGULAR  SYSTEM STORE THE
              SOLUTION TEMPORARILY IN ZZ( )
      ASSIGN 200 TO NEXT
      GO TO 400
  200 CONTINUE
        * * * SECONDARY LOOP BEGINS HERE * * * ITER
C       ATION COUNTER
  210 ITER=ITER+1
      IF(ITE.LE.ITMAX) GO TO 220
      MODE=3
      WRITE(6, 440)
      GO TO350
  220 CONTINUE
      ALPHA=TWO
      DO 240 IP=1, NSETP
      L=INDEX(IP)
      IF(ZZ(IP))230, 230, 240
  230 T= -X(L)/(ZZ(IP)-X(L))
      IF(ALPHA.LE.T) GO TO 240
      ALPHA=T
      JJ=IP
  240 CONTINUE
C           IF ALL NEW CONSTRAINED COEFFS ARE FE-
                ASIBLE THEN ALPHA WILL
            STILL=2.IF EXIT FROM SECONDARY LOOP TO
C               MAIN LOOP
      IF(ALPHA.EQ.TWO)  GO  TO 330
      DO 250 IP=1, NSETP
      L=INDEX(IP)
  250 X(L)=X(L)+ALPHA*(ZZ(IP)-X(L))
C      SET P TO SET  Z
      I=INDEX(JJ)
```

```
260  X(J)=ZERO
     IF(JJ.EQ.NSETP)  GO  TO 290
     JJ=JJ+1
     DO 280  J=JJ, NSETP
     II=INDEX(J)
     INDEX(J-1)=II
     CALLG1(A(J-1, II), A(J, II), CC, SS, A(J-1,  II)
     A(J, II)=ZERO
     DO 270 L=1, N
     IF(L.NE.II)  CALLG2(CC, SS, A(J-1, L), A(J, L
270  CONTINUE
280  CALL G2(CC, SS, B(J-1), B(J))
290  NPP1=NSETP
     NSETP=NSETP-1
     IZ1=IZ1-1
     INDEX(IZ1)=I
     DO300 JJ=1, NSETP
     I=INDEX(JJ)
     IF(X(I))  260, 260, 300
300  CONTINUE
     DO 310 I=1, M
310  ZZ(I)=B(I)
     ASSIGN 320 TO NEXT
     GO TO400
320  CONTINUE
     GO TO 210
C       * * *END OF SECONDARY LOOP * * *
330  DO 340 IP=1, NSETP
     I=INDEX(IP)
340  X(I)=Z(IP)
C       ALL NEW COEFFS ARE POSITIVE.LOOP BACK TO
C       BEGINNING
```

```
      GO TO 30
C             * * *END OF MAIN LOOP* * *
  350 SM=ZERO
      IF(NPP1.GT.M)  GO  TO  370
      DO 360 I=NPP1, M
  360 SM=SM+B(I) * * 2
      GOTO 390
  370 DO 380  J=1, N
  380 W(J)=ZERO
  390 RNORM=SQRT(SM)
      RETURN
C      THE FOLLOWING  BLOCK OF CODE IS USED AS
          AN INTERNAL SUBROUTINE
C      TO SOLVE THE TRIANGULAR SYSTEM.PUTTING
          THE SOLUTION IN ZZ(  )
  400 DO 430 L=1, NSETP
      IP=NSETP+1-L
      IF(L.EQ.1)  GO  TO 420
      DO 410  II=1, IP
  410 ZZ(II)=ZZ(II)-A(II, JJ)*ZZ(IP+1)
  420 JJ=INDEX(IP)
  430 ZZ(IP)=ZZ(IP)/A(IP, JJ)
      GO TO NEXT, (200, 320)
  440 FORMAT(35HO NNLS QUITTING ON ITERATION
     \   COUNT)
      END
      SUBROUTINE H12(MODE, LP, L1, M, U, IUE, UP,
        C, ICE, ICV, NCV)
C     CONSTRUCTION AND/OR APPLICATION OF A SI-
C     NGLE HOUSEHOLDER
C     TRANSFORMATION..        Q=I+U*(U* *T)/B
      DIMENSIONU(IUE, M), C(1)
```

```
      DOUBLE PRECISION SM, B
      ONE=1.0
      IF(0.GE.LP. OR.LP.GE.L1.OR.L1.GT.M) RETURN
      CL=ABS(U(1, LP))
      IF(MODE.EQ.2)GOTO60
      DO 10 J=L1, M
   10 CL=AMAX1(ABS(U(1, J)), CL)
      IF(CL)130, 130,20
   20 CLINV=ONE/CL
      SM=(DBLE(U(1, LP))*CLINV)* *2
      DO 30 J=L1, M
   30 SM=SM+(DBLE(U(1, J))*CLINV)* *2
      SM1=SM
      CL=CL*SQRT(SM1)
      IF(U(1, LP))50, 50, 40
   40 CL=-CL
   50 UP=U(1, LP)-CL
      U(1, LP)=CL
      GO TO 70
C        * * APPLY THE TRANSFORMATION  I+U*(U* *T)
C        /B TO C    * *
   60 IF(CL)130, 130, 70
   70 IF(NCV.LE.0)RETURN
      B=DBLE(UP)*U(1, LP)
      IF(B)80, 130, 130
   80 B=ONE/B
      I2=1-ICV+ICE*(LP-1)
      INCR=ICE*(L1-LP)
      DO120  J=1, NCV
      I2=I2+ICV
      I3=I2+INCR
      I4=I3
```

```
      SM=C(I2)*DBLE(UP)
      DO 90 I=L1, M
      SM=SM+C(I3)*DBLE(U(1, I))
 90   I3=I3+ICE
      IF(SM) 100, 120, 100
100   SM=SM*B
      C(I2)=C(I2)+SM*DBLE(UP)
      DO 110  I=L1, M
      C(I4)=C(I4)+SM*DBLE(U(1, I))
110   I4=I4+ICE
120   CONTINUE
130   RETURN
      END
      SUBROUTINE  G1(A, B, C, S, R)
C          COMPUTE  ORTHOGONAL  ROTATION  MATRIX
      ZERO=0.0
      ONE=1.0
      IF(ABS(A).LE.ABS(B))  GO  TO10
      XR=B/A
      YR=SQRT(ONE+XR* *2)
      C=SIGN(ONE/YR, A)
      S=C*XR
      R=ABS(A)*YR
      RETURN
 10   IF(B)20, 30, 20
 20   XR=A/B
      YR=SQRT(ONE+XR* *2)
      S=SIGN(ONE/YR, B)
      C=S*XR
      R=ABS(B)*YR
      RETURN
 30   R=ZERO
```

```
      C=ZERO
      S=ONE
      RETURN
      END
      SUBROUTINE  G2(C, S, X, Y)
C        APPLY THE ROTATION COMPUTED BY G1 TO
C        (X, Y)
      XR=C*X+S*Y
      Y=-S*X+C*Y
      X=XR
      RETURN
      END
      FUNCTION DIFF (X, Y)
      DIFF=X-Y
      RETURN
      END
```

对于上述的数值例子，其计算结果及打印格式如下：

```
EXAMPLE OF CONSTRAINED CURVE FITTING US-
  INGTHE SUBROUTINE LDP.
RELATED INTERMEDIATE QUANTITIES ARE GRV-
  EN.
V=           -0.46711      0.88420
             -0.88420     -0.46711
F TILDA= -1.53602   0.38402   -0.05353   0.17408
S=      2.25455              0.34571
G TILDA= -0.20718       2.55762
             -0.39219     -1.35115
              0.59937     -1.20647
H TILDA= -1.30041     -0.08354     0.38395
MODE (FROM LDP)=1     ZNORM=0.28501
Z=          0.12680      -0.25525
THE COEFICIENTS OF THE FITTED LINE F(T)=X
```

```
(1)*T+X(2) ARE
X(1)=      0.62131     AND X(2)=      0.37869
THE CONSECUTIVE RESIDUALS ARE
      1     -0.03401     2  -0.08934
      3      0.01066     4   0.32426
THE RESIDUALS NORM IS        0.33823
```

## § 3-6 带惩罚函数的阻尼最小二乘算法程序和实例

本程序用于解约束条件下(特别是线性不等式约束)的地球物理反演问题。方法原理已在 § 3-3 中叙述。本程序是按照一个位場反演例子编写的。用于其它问题时应作适当改变。本程序用FORTRAN-Ⅳ语言编写，包括四个子程序及一个主程序。

### (一) 主要标识符

1. 主程序

I，J，K——循环控制变量；

M——参加反演的采样点数(观测点数)；

N——地球物理模型的参量数；

IH——约束方程个数；

DX——采样间隔或测点距(等间隔)；

EC——用实际资料或理论模型作反演的控制变量；

$$EC=\begin{cases}0 & \text{实际资料反演；}\\ 1 & \text{理论模型反演。}\end{cases}$$

XM——理论模型的参量值数组；

XO——反演模型参量初始值数组；

GE——包括以上DX，EC，XO等参量的数组；

G，C——约束方程所构成的系数矩阵及右端列向量；

GM——实际地球物理观测值数组；

FUN1—— 迭代过程中每一循环计算的理论值数组。

**2. 子程序XASU(X, FUN, M, N)**

计算模型体正演问题理论值的子程序。一般应根据具体的地球物理问题自行编制。本例是计算二维有限延伸板状体的磁异常值。其中

X——模型体参量数组；

FUN——地球物理理论值数组。

**3. 子程序TEDU(X, GAC, FUN0, M, N)**

求地球物理理论值对各参量的偏导数，形成Jacobi矩阵。偏导数的计算采用扰动法，步长为 $1.0\times10^{-5}$。其中

GAC——Jacobi 矩阵数组　其余同上。

**4. 子程序QF(F, DG, B, GM, FUN, M)**

求地球物理理论值与实测值(或理论模型试验值)偏差平方和。其中

F——偏差平方和；

DG——各点偏差值数组。其余同上

**5. 子程序MAQT(D0, B0, X0, V, FUN, GM, G, C, M, N, IH)**

带惩罚函数的阻尼最小二乘法计算过程。

其中虚元

$D_0$——初始阻尼因子 $\nu_0$；

$X_0$——参量的初始数组；

V——初始惩罚因子。其余同上。

该子程序中还有一些主要的标识符，现说明如下：

FL——确定约束满足与否的控制数组；

E——惩罚项构成的矩阵；

A——前n行n列为正则方程系数矩阵，第n+1列为方程组的右端项；

Q——迭代过程中每一步的改正量；

R——A矩阵对角元素平方和；

X——迭代过程参量的改正结果。

本子程序迭代结束标准为满足下列两个不等式之一：

$|F0-F1|\leqslant EW$，本程序$EW$取$10^{-3}$；

迭代次数　IDAD＞10

## （二）程序的使用说明

输入按下列顺序进行：

（1）控制参量和初始模型参量数组GE；

（2）约束方程组系数矩阵数组 G；

（3）约束方程组右端项数组 C；

（4）理论模型参量数组XM；

（5）实测地球物理观测值数组 GM。

计算结果按下列顺序打印：

（1）输出全部初始输入数据；

（2）输出每次迭代的次数 IDAD，偏差平方和FO，阻尼及惩罚因子值等；

（3）输出反演结果数组X，以及反演计算异常数组FUN。

## （三）反演计算实例

本程序给出用惩罚函数法进行反演计算的一个位场反演实例。选用二维斜磁化板状体模型。该模型体共有七个不同的参量：

$x_0$——模型中心点横坐标　（单位：m）

$h$——模型中心点纵坐标　（单位：m）

$2b$——模型体宽度　（单位：m）

$2l$——模型体延深长度　（单位：m）

$a$——模型体倾角　（单位：(°)）

$\theta$——磁化方向倾角　（单位：(°)）

$J$——磁化强度　（单位：$10^{-6}$CGSM）

理论模型的参量选择为

$$x(1)=x_0=160\text{m},\qquad x(2)=h=60\text{m},$$
$$x(3)=2b=20\text{m},\qquad x(4)=2l=50\text{m},$$
$$x(5)=\alpha=120^\circ,\qquad x(6)=\theta=90^\circ,$$
$$x(7)=J=1500\times10^{-6}\text{CGSM}.$$

由图3-6-1可见，若采用不带约束条件的阻尼最小二乘法反演，尽管反演计算得到的位場曲线与理论计算曲线符合得很

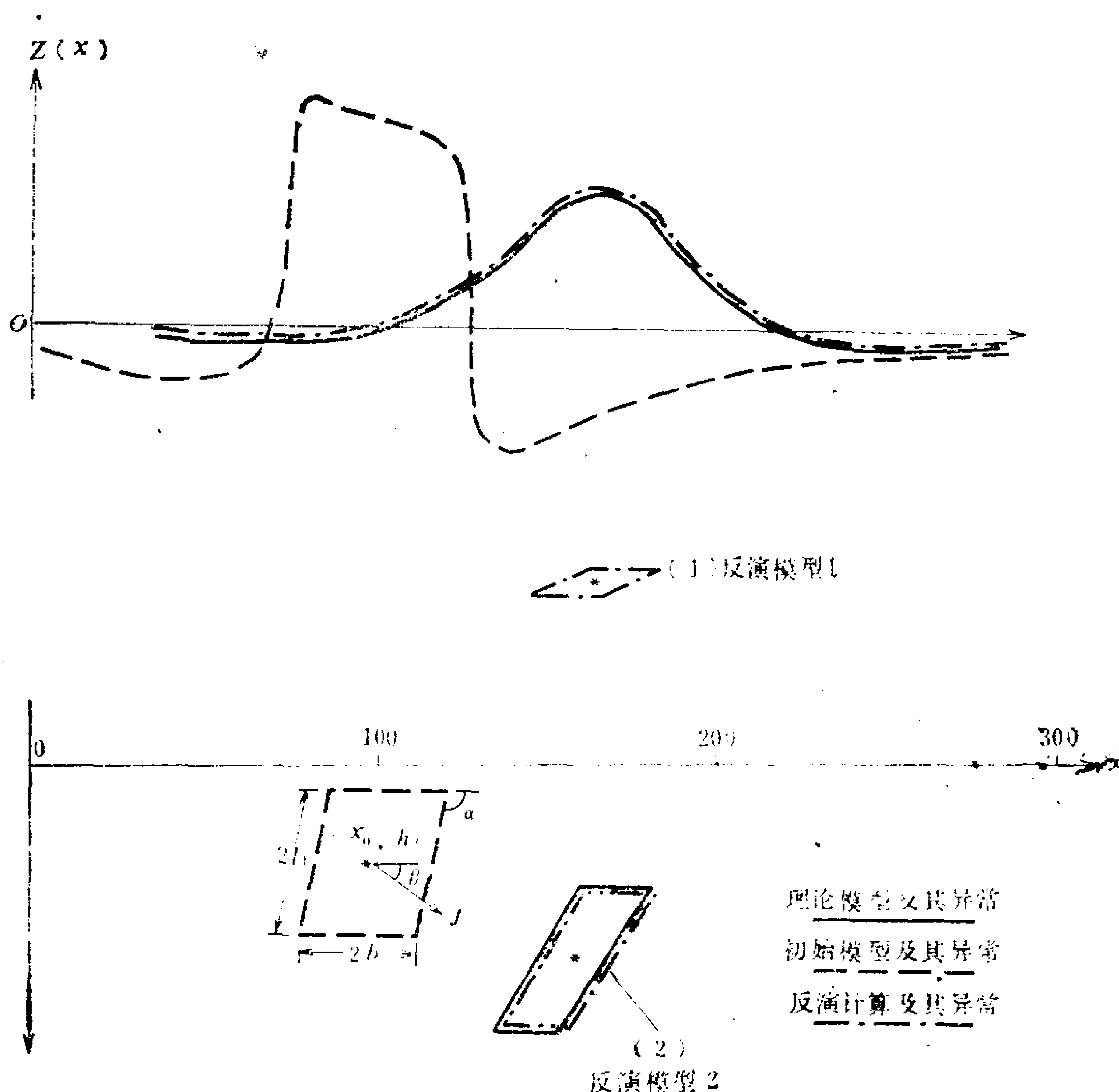



图3-6-1

好，但得到的模型体与原设模型有很大差异（负的深度，见反演模型1）如采用约束条件反演，从同一初始模型出发，能很快收敛于正确结果（反演模型2）。约束方程组为

$x(1)\leqslant 200$, $x(2)\leqslant 100$, $x(4)-2x(3)\leqslant 20$,

$x(5)\leqslant 120$, $x(6)\leqslant 100$, $x(7)\leqslant 1600$

约束方程组的矩阵形式为

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} \leqslant \begin{pmatrix} 200 \\ 100 \\ 20 \\ 120 \\ 100 \\ 1600 \end{pmatrix}$$

## （四） FORTRAN-Ⅳ语言的源程序

```
C           * *     MAIN  PROGRAM     * *
      IMPLICIT  REAL*8(A-H, O-Z)
      DIMENSION  GE(10), G(6, 7), C(6)
      DIMENSION  XM(7), X0(7), FUN1(30), GM(30)
      COMMON  XK(30), ZK(30)
      M=30
      N=7
      IDIT=0
      V=1.0
      B=0.0
      IH=6
      B=0.0
      READ(5, *) (GE(I), I=1, 10)
      WRITE(6, 20) GE
   20 FORMAT(5H0GE=, 10F10.3)
      READ (5, *) ((G(I, J), J=1, N), I=1, IH)
      WRITE(6, 30) ((G(I, J), J=1, N), I=1, IH)
   30 FORMAT (5H0BE=, 7F10.3/(5X, 7F10.3))
      READ (5, *) (C(I), I=1, IH)
```

```
      WRITE(6, 40)  C
 40   FORMAT(5H0  C=, 7F10.3)
      DX=GE(1)
      DC=GE(2)
      EC=GE(3)
      DO 45 I=1, M
 45   XK(I)=(I-1)*DX
      IF(DC) 70, 70, 50
 50   READ(5, *)  (ZK(I), I=1, M)
 70   DO 80 I=1, M
 80   ZK(I)=0.0
 90   DO 100 I=1, N
100   X0(I)=GE(3+I)
      IF(EC)130, 130, 110
110   READ(5, *)(XM(I), I=1, N)
      WRITE(6, 120)XM
120   FORMAT(5H0XM=, 7F10.3)
      CALL XASU(XM, GM, M, N)
      GO TO 140
130   READ(5, *)  (GM(I), I=1, M)
140   WRITE  (6, 150)(GM(I), I=1, M)
150   FORMAT(5X, 10F10.3/(5X, 10F10.3))
200   IDIT=IDIT+1
      IF(IDIT.GT.10)  GO TO 260
      CALL MAQT(1.0D-1, B, X0, V, FUN1, GM, G, C, M,
     N, IH)
      DO  220  J=1, N
      IF(X0(J))  210, 220, 220
210   X0(J)= -X0(J)
220   CONTINUE
      DO 250 I=1, IH
      US=0.0
```

```
      DO  230  J=1, N
 230  US=US+G(I, J)*X0(J)
      IF(DABS((US-C(I))/C(I)).LT.0.1)GOTO250
 240  V=5.0*V
      GO TO 200
 250  CONTINUE
 260  WRITE(6, 270)  IDIT, V, X0
 270  FORMAT(1HO, 'IDIT=' I2, 2X, 'V=', F15.1, 2X,
      'XO=', 7E12.3)
      WRITE(6, 170)  (FUN1(I), I=1, M)
 170  FORMAT(6H0FUN1=, 10F12.3/(6X, 10F12.3))
 300  STOP
      ENd
        ** THIS SUBROUTINE MINIMIZES THE OBJECTI-
C          VE FUNCTION USING
C        THE CONSTRAINED AND DAMPED LEAST SQUA
C        RES METHOD **
      SUBROUTINEMAQT(D0, B0, X0, V, FUN, GM, G, C,
      M, N, IH)
      IMPLICIT   REAL*8(A-H, O-Z)
      DIMENSION X0(N), G(IH, N), C(IH), FUN(M), GM-
        (M)
      DIMENSION FL(6), G2(7, 6), X(7), B(8)
      DIMENSION GAC(30, 8), A(8, 8), DG(30), CB(7), GB-
        (7), E(7, 7), R(8), Q(8)
      D=D0
      IDAD=0
      IB=N+1
      EW=1.0D-3
      CALL XASU(X0, FUN, M, N     )
      CALL QF(F0, DG, B0, GM, FUN, M)
      WRITE(6, 270) F0, D, X0
```

```
30     D=D/10.
       IDAD=IDAD+1
       WRITE(6, *) IDAD
       DO 50 I=1, IH
       US=0.0
       DO 35 J=1, N
35     US=US+G(I, J)*X0(J)
       IF(DABS(US-C(I))-5.0) 45, 45, 40
40     FL(I)=1.0
       GOTO 50
45     FL(I)=0.0
50     CONTINUE
       CALL TEDU(X0, GAC, FUN, M, N)
       DO 60 I=1, M
60     GAC(I, IB)=DG(I)
       DO 80 I=1, N
       DO 80 J=1, IB
       A(I, J)=0.0
       DO 70 K=1, M
70     A(I, J)=A(I, J)+GAC(K, I)*GAC(K, J)
80     CONTINUE
       DO 100 I=1, N
       HF=0.0
       DO 90 J=1, IH
       G2(I, J)=FL(J)*G(J, I)
90     HF=HF+G2(I, J)*C(J)
       CB(I)=HF
100    CONTINUE
       DO 120 I=1, N
       DO 120 K=1, N
       HF=0.0
       DO 110 J=1, IH
```

```
110 HF=HF+G2(I, J)*G(J, K)
    E(I, K)=HF
120 CONTINUE
    DO 140 I=1, N
    GB(I)=0.0
    DO 130 J=1, N
    A(I, J)=A(I, J)+V*E(I, J)
130 GB(I)=GB(I)+E(I, J)*X0(J)
    GB(I)=V*(GB(I)-CB(I))
    A(I, IB)=A(I, IB)-GB(I)
    R(I)=DSQRT(A(I, I))
    IF(R(I).EQ.0.)  WRITE(6, *)  R(I)
140 CONTINUE
    DO 160 I=1, N
    IG=I+1
    IF(IG.GT.N)     GO TO 155
    DO 150 J=IG, N
150 A(I, J)=A(I, J)/R(I)/R(J)
155 A(I, IB)=A(I, IB)/R(I)
160 CONTINUE
165 DO 200 I=1, N
    X(I)=D+1.0
    IG=I-1
    IF(IG.LT.1) GO TO 170
    DO 170 K=1, IG
    B(K)=X(K)*A(I, K)
    X(I)=X(I)-B(K)*A(I, K)
170 CONTINUE
    IE=I+1
    DO  190 J=IE, IB
    A(J, I)=A(I, J)
    IF(IG.LT.1) GO TO 190
```

```
      DO 180 K=1, IG
180   A(J, I)=A(J, I)-B(K)*A(J, K)
190   A(J, I)=A(J, I)/X(I)
200   CONTINUE
      DO 220 I=1, N
      K=N-I+1
201   L=K+1
      IF(L.GT.N)  GO TO 202
      DO 210 J=L, N
210   A(IB, K)=A(IB, K)-A(J, K)*A(IB, J)
202   Q(K)=A(IB, K)/R(K)
      X(K)=X0(K)+Q(K)
      IF(X(K)) 205, 220, 220
205   X(K)= -X(K)
220   CONTINUE
      CALL XASU(X, FUN, M, N     )
      CALL QF(F1, DG, B0, GM, FUN, M)
      IF(F1-F0)  250, 250, 280
250   S=(F0-F1)/F1
      F0=F1
      DO 260 I=1, N
260   X0(I)=X(I)
      WRITE(6, 270) F0, D, X0
270   FORMAT(1H0, 'F0=', E15.5, 2X, 'D=', F14.7, 2X,
        'X0=', 7F12.3/(25X, 7F12.3))
      IF(IDAD.LT.10.AND.S.GT.EW)GOTO 30
      GO TO 300
280   D=D*10.0
      IF(D.GT.1.0D8) GO TO 300
      GO TO 165
300   WRITE(6, 310)    FUN
310   FORMAT(5H0FUN=, 10F12.3/(5X, 10F12.3))
```

```
      RETURN
      END
C         * * COMPUTE MODEL FUNCTION F(X) * *
      SUBROUTINE XASU(X, FUN, M, N)
      IMPLICIT   REAL * 8(A-H, O-Z)
      DIMENSION X(N), FUN(M)
      COMMON   XK(30), ZK(30)
      PI=3.14159265D0/180.0
      AA=DSIN(X(5)*PI)
      BB=DCOS(X(5)*PI)
      CC=DSIN(X(6)*PI)
      DD=DCOS(X(6)*PI)
      DO 10 K=1, M
      A=X(2)-ZK(K)+X(4)*AA/2.0
      B=X(2)-ZK(K)-X(4)*AA/2.0
      C=XK(K)-X(1)+X(4)*BB/2.0
      D=XK(K)-X(1)-X(4)*BB/2.0
      E=C+X(3)/2.0
      F=C-X(3)/2.0
      G=D+X(3)/2.0
      H=D-X(3)/2.0
      U=(B*B+F*F)*(A*A+G*G)/(B*B+E*E)/(A*A+
         H*H)
      V=X(3)*B/(B*B+C*C-X(3)*X(3)/4.0)
      W=X(3)*A/(A*A+D*D-X(3)*X(3)/4.0)
      U=DLOG(U)
      V=DATAN(V)
      W=DATAN(W)
      FUN(K)=2.0*X(7)*AA*((DD*AA-CC*BB)
      /2.0*U+(DD*BB+CC*AA)*(V-W))
 10   CONTINUE
300   RETURN
```

```
      END
C       • • THIS SUBROUTINE COMPUTES THE SUM OF
C         THE  DEVIATION
C           SQUARE VALUES OF FUNCTIONS• •
      SUBROUTINE QF (F, DG, B, GM, FUN, M)
      IMPLICIT    REAL*8(A-H, O-Z)
      DIMENSION DG(M), GM(M), FUN(M)
      F=0.0
      DO 1000 K=1, M
      DG(K)=GM(K)-FUN(K)-B
      F=F+DG(K)*DG(K)
 1000 CONTINUE
      RETURN
      END
C           * * COMPUTE    JACOBI    MATRIX    GAC * *
C       SUBROUTINE TEDU(X, GAC, FUN0, M, N)
      IMPLICIT    REAL*8(A-H, O-Z)
      DIMENSION    X(N), GAC(M, N), FUN(30), FUN0(M
      DO 11 K=1, M
   11 FUN(K)=FUN0(K)
      DO 13I=1, N
      X(I)=X(I)+1D-5
      CALL XASU(X, FUN0, M, N)
      X(I)=X(I)-1D-5
      DO  12 J=1, M
   12 GAC(J, I)=(FUN0(J)-FUN(J))/1D-5
   13 CONTINUE
      RETURN
      END
```

## §3-7 可变容限算法程序和实例

本程序用于解一般非线性规划问题。使用了单纯形加速法计

算目标函数极小。该程序用 FORTRAN-Ⅳ 语言编写，包括五个子程序和一个主程序。参数的信息传递全部由公共区完成。

## （一）主要标识符

I、J、K——循环控制变量；

NX——参量总数；

NC——等式约束总数；

NIC——不等式约束总数；

STEP——单纯形多面体的大小$t$；

CONVER——结束搜索的收敛标准$\varepsilon$；

ALFA——反射系数$\alpha$；

BETA——收缩系数$\beta$；

GAMA——扩张系数$\gamma$；

ICONT——记录迭代次数的计数单元；

NCONT——打印每一$(NX+1)$步的计数单元；

FDIFER——违反约束的容限标准$\phi$；

LOW——目标函数最低值指标；

LHIGH——目标函数最大值指标；

LSEC——目标函数第二最大值指标；

X——初始搜索点；

X1，X2——多面体各顶点坐标数组；

R——目标函数及等式不等式约束的值的数组。

以下为各子程序的功能：

（1）FEASBL　求$T(x)$极小以寻求可行或非可行点的子程序。通常当违反约束情况的组合值大于容限标准时被调用一次。

（2）START　计算初始多面体各顶点坐标的子程序。

（3）SUMR　计算违反约束方程的值的平方和，以便于与容限标准相比较。

（4）WRITEX　输出打印目标函数，等式、不等式约束和结

果值的子程序。

(5) PROB(INQ) 把要解的问题引入到计算程序中。其中虚元

$$INQ=\begin{cases}1 & 表示等式约束,\\ 2 & 表示不等式约束,\\ 3 & 表示目标函数。\end{cases}$$

## (二) 程序的使用

1. 子程序PROB(INQ)的组织

(1) 在说明语句"EQUALITY CONSTRAINTS"之后写入等式约束

$$\begin{aligned}1\quad & R(1)=h_1(x),\\ & \cdots\cdots\qquad ,\\ & R(m)=h_m(x).\end{aligned}$$

(2) 在说明语句"INEQUALITY CONSTRAINTS"之后写入不等式约束

$$\begin{aligned}2\quad & R(m+1)=g_{m+1}(x),\\ & \cdots\cdots\qquad ,\\ & R(p)=g_D(x).\end{aligned}$$

(3) 在说明语句"OBJECTVE FUNCTION"之后写入目标函数

$$3\quad R(p+1)=\Phi(x).$$

如果问题不包含等式约束($m=0$),则在标号1后不写入任何语句,且$R(m+1)$变为$R(1)$。同样如有等式而无不等式约束标号2后面略去任何语句,使$R(p+1)=R(m+1)$;在无约束情况下$R(p+1)=R(1)$。

2. 第一批输入参量NX,NC,NIC,SIZE,CONVER等其中CONVER$=\varepsilon$可取为$10^{-5}$或$10^{-6}$。SIZE$=t_0$搜索最初阶段多面体的大小,当$x$上下界确定后,则:

(1) 若$x$期望的变化范围沿各坐标相同，SIZE取20%$x$的上下界之差。

(2) 若$x$期望的变化范围沿各坐标不同，则取SIZE为上下界之差最小的值。

(3) 第二批输入初始点坐标$x^{(0)}$值。

## (三) 计 算 实 例

用本程序计算了以下非线性规划问题：

$$\min \Phi(x) = 4x_1 - x_2^2 - 12.$$

满足约束条件

$$h_1(x) = 25 - x_1^2 - x_2^2 = 0,$$

$$g_2(x) = 10x_1 - x_1^2 + 10x_2 - x_2^2 - 34 \geqslant 0,$$

$$g_3(x) = x_1 \geqslant 0,$$

$$g_4(x) = x_2 \geqslant 0.$$

当初值取为 $x^{(0)} = (1,1)$，$t = 0.3$，$\phi^{(0)} = 2(m+1)t = 1.2$ 时，其搜索轨迹及迭代次数如图3-7-1所示。最后计算的结果为

$$x^* = (1.001,\ 4.989),$$

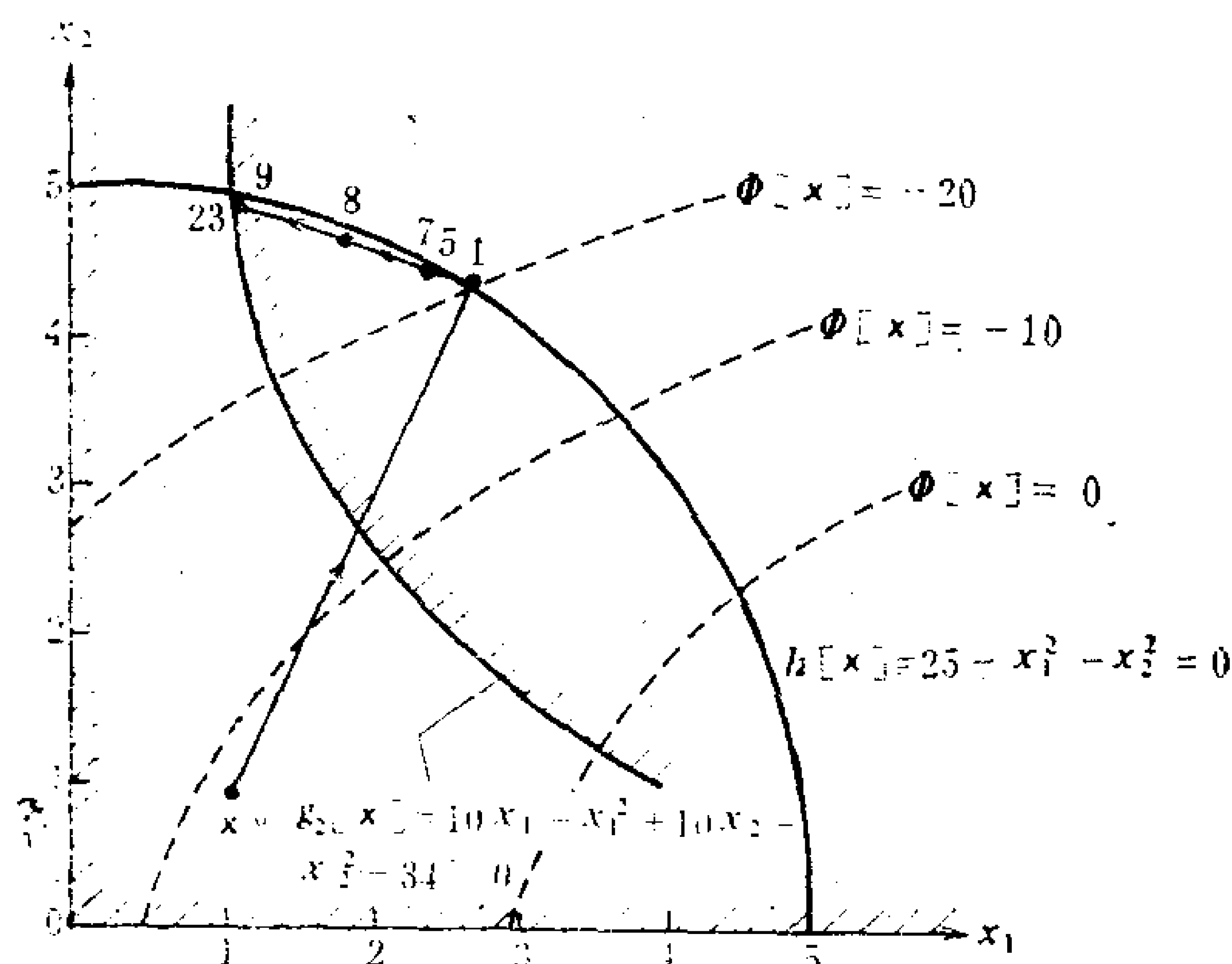



图3-7-1

$\Phi(x^*) = -31.992.$

## （四）FORTRAN-Ⅳ语言源程序

```
C              * * * PROGRAM    FLEXIPLEX * * *
          DIMENSION X(50), X1(50, 50), X2(50, 50), R(100),
             SUM(50), F(50), SR(50),
             1ROLD(100), H(50)
          COMMON/A/NX, NC, NIC, STEP, ALFA, BETA, GA-
             MA, IN, INF, FDIFER, SEQL, K1, K2,
          K3, K4, K5, K6, K7, K8, K9, X, X1, X2, R, SUM,
             F, SR, ROLD, SCALE, FOLD
          COMMON/B/LFEAS, L5, L6, L7, L8, L9, R1A, R2A,
             R3A
C         PROBLEM IDENTIFICATION HEADER IS READ IN
C            AFTER THIS CARD
C          PARAMETERS FOR THE PROBLEM ARE READ IN
C            AFTER THIS CARD
          READ(5, 1)  NX, NC, NIC, SIZE, CONVER
          ALFA=1.
          BETA=0.5
          GAMA=2.
          TIME=1.
C         PERMANENT DATA FOR THE PROBLEM SHOULD
C           BE READ IN AFTER THIS CARD
C         TEMPORARY  DATA FOR THE PROBLEM, SUCH AS
C            VARIABLE COEFFICIENTS OR
C         NEW  PARAMETERS SHOULD BE READ IN AFTER
C            THIS CARD
          STEP=SIZE
C         THE ASSUMED INITIAL VECTOR IS READ IN AF-
C            TER THIS CARD
          READ(5, 2)   (X(I), I=1, NX)
```

```
11   WRITE(6, 106)
     WRITE(6, 759)
     WRITE(6, 756)  NX,NC,NIC, SIZE, CONVER, TIME
     K1 = NX + 1
     K2 = NX + 2
     K3 = NX + 3
     K4 = NX + 4
     K5 = NX + 5
     K6 = NC + NIC
     K7 = NC + 1
     K8 = NC + NIC
     K9 = K8 + 1
     N = NX - NC
     N1 = N + 1
     IF(N1.GE.3) GO  TO  50
     N1 = 3
     N = 2
50   N2 = N + 2
     N3 = N + 3
     N4 = N + 4
     N5 = N + 5
     N6 = N + 6
     N7 = N + 7
     N8 = N + 8
     XN = N
     XNX = NX
     XN1 = N1
     R1A = 0.5 * (SQRT(5.) - 1.)
     R2A = R1A * R1A
     R3A = R2A * R1A
     L5 = NX + 5
     L6 = NX + 6
```

```
      L7 = NX + 7
      L8 = NX + 8
      L9 = NX + 9
      ICONT = 1
      NCONT = 1
      WRITE(6, 115)
      WRITE(6, 116)  (X(J), J=1, NX)
      FDIFER = 2. * (NC+1) * STEP
      FOLD = FDIFER
      IN = N1
      CALLSUMR
      SR(N1) = SQRT(SEQL)
      WRITE(6, 763)  FDIFER, SR(N1)
      IF(SR(N1).LT.FDIFER)  GO  TO  341
      CALL  WRITEX
      WRITE(6, 757)
      INF = N1
      STEP = 0.05 * FDIFER
      CALL  FEASBL
      WRITE(6, 764)
      WRITE(6, 116)  (X2(INF, J), J=1, NX)
      WRITE(6, 765)  SR(INF)
      IF(FOLD.LT.1.0E-9)  GO  TO  80
  341 WRITE(6, 35)
      WRITE(6, 758)   ICONT, FDIFER
      CALL  WRITEX
      FTER = R(K9)
C     COMPUTE CENTROID OF ALL VERTICES OF INIT-
C       IAL POLYHEDRON
  237 STEP1 = STEP * (SQRT(XNX+1.)+XNX-1.)/(XNX *
        SQRT(2.))
      STEP2 = STEP * (SQRT(XNX+1.)-1.)/(XNX * SQRT
```

```
      .(2.))
      ETA=(STEP1+(XNX-1.)*STEP2)/(XNX+1.)
      DO 4 J=1, NX
      X(J)=X(J)-ETA
    4 CONTINUE
      CALL START
      DO 9 I=1, N1
      DO 9 J=1, NX
      X2(I, J)=X1(I, J)
    9 CONTINUE
      DO 5 I=1, N1
      IN=I
      DO 6 J=1, NX
    6 X(J)=X2(I, J)
      CALL SUMR
      SR(I)=SQRT(SEQL)
      IF(SR(I).LT.FDIFER) GO TO 8
      CALL FEASBL
      IF(FOLD.LT.1.0E-9) GO TO 80
    8 CALL PROB(3)
      F(I)=R(K9)
    5 CONTINUE
 1000 STEP=0.05*FDIFER
      ICONT=ICONT+1
C     SELECT LARGEST VALUE OF OBJECTIVE FUNCT-
C     ION FROM POLYHEDRON VERTICE
      FH=F(1)
      LHIGH=1
      DO 16 I=2, N1
      IF(F(I).LT.FH) GO TO 16
      FH=F(I)
      LHIGH=I
```

```
16  CONTINUE
    SELECT MINIMUM VALUE OF OBJECTIVE FUNCTI-
C     ON FROM POLYHEDRON VERTICE
41  FL=F(1)
    LOW=1
    DO 17 I=2, N1
    IF(FL.LT.F(I)) GO TO 17
    FL=F(I)
     LOW=I
17  CONTINUE
    DO 86 J=1, NX
86  X(J)=  X2(LOW, J)
    IN=LOW
    CALL SUMR
    SR(LOW)=SQRT(SEQL)
    IF(SR(LOW).LT.FDIFER) GO TO 87
    INF=LOW
    CALL FEASBL
    IF(FOLD.LT.1.0E-09) GO TO 80
    CALL PROB(3)
    F(LOW)=R(K9)
    GO TO 41
 87 CONTINUE
C     FIND CENTROID OF POINTS WITH I DIFFERENT
C        THAN LNIGH
    DO  19 J=1, NX
    SUM2=0.
    DO 20 I=1, N1
20  SUM2=SUM2+X2(I, J)
19  X2(N2, J)=1./XN*(SUM2-X2(LHIGH, J))
    SUM2=0.
    DO 36 I=1, N1
```

```
      DO 36 J = 1, NX
      SUM2 = SUM2 + (X2(I, J) - X2(N2, J)) * * 2
   36 CONTINUE
      FDIFER = (NC + 1)/XN1 * SQRT(SUM2)
      IF(FDIFER.LT.FOLD) GO TO 98
      FDIFER = FOLD
      GO TO 198
   98 FOLD = FDIFER
  198 CONTINUE
      FTER = F(LOW)
  137 NCONT = NCONT + 1
      IF(NCONT.LT.4 * N1) GO TO 37
      IF(ICONT.LT.1500) GO TO 337
      FOLD = 0.5 * FOLD
  337 NCONT = 0
      WRITE(6, 35)
      WRITE(6, 758) ICONT, FDIFER
      CALL  WRITEX
   37 IF(FDIFER.LT.CONVER) GO TO 81
C      SELECT SECOND LARGEST VALUE OF OBJECTIVE
C         FUNCTION
      IF(LHIGH.EQ.1)GOTO43
      FS = F(1)
      LSEC = 1
      GO TO 44
   43 FS = F(2)
      LSEC = 2
   44 DO 18 I = 1, N1
      IF(LHIGH.EQ.I) GO TO 18
      IF(F(I).LT.FS) GO TO 18
      FS = F(I)
      LSEC = I
```

```
   18  CONTINUE
 C      REFLECT HIGH POINT THROUGH CENTROID
       DO 61 J=1, NX
       X2(N3, J)=X2(N2, J) +ALFA*(X2(N2, J)-X2(LHI-
         GH, J))
   61  X(J)=X2(N3, J)
       IN=N3
       CALL SUMR
       SR(N3)=SQRT(SEQL)
   89  IF(SR(N3).LT.FDIFER) GO TO 82
       INF=N3
       CALL FEASBL
       IF(FOLD.LT.1.0E-9) GO TO 80
   82  CALL  PROB(3)
       F(N3)=R(K9)
       IF(F(N3).LT.F(LOW)) GO TO 84
       IF(F(N3).LT.F(LSEC)) GO TO 92
       GO TO 60
   92  DO 93 J=1, NX
   93  X2(LHIGH, J)=X2(N3, J)
       SR(LHIGH)=SR(N3)
       F(LHIGH)=F(N3)
       GO TO 1000
 1      EXPAND VECTOR OF SEARCH ALONG DIRECTION
 C         THROUGH CENTROID AND
 C      REFLECTED VECTOR
   84  DO 23 J=1, NX
       X2(N4, J)=X2(N3, J)+GAMA*(X2(N3, J)-X2(N2, J))
   23  X(J)=X2(N4, J)
       IN=N4
       CALLSUMR
       SR(N4)=  SQRT(SEQL)
```

```
      IF(SR(N4).LT.FDIFER) GO TO 25
      INF = N4
      CALLFEASBL
      IF(FOLD.LT.1.0E-09) GO TO 80
 25   CALL PROB(3)
      F(N4) = R(K9)
      IF(F(LOW).LT.F(N4))GOTO 92
      DO 26 J=1, NX
 26   X2(LHIGH, J) = X2(N4, J)
      F(LHIGH) = F(N4)
      SR(LHIGH) = SR(N4)
      GO TO 1000
 60   IF(F(N3).GT.F(LHIGH)) GO TO 64
      DO 65 J=1, NX
 65   X2(LHIGH, J) = X2(N3, J)
 64   DO 66 J=1, NX
      X2(N4, J) = BETA * X2(LHIGH, J) + (1.- BETA) * X2(N-
         2, J)
 66   X(J) = X2(N4, J)
      IN = N4
      CALLSUMR
      SR(N4) = SQRT(SEQL)
      IF(SR(N4).LT.FDIFER) GO TO 67
      INF = N4
      CALL FEASBL
      IF(FOLD.LT.1.0E-9) GO TO 80
 67   CALL PROB(3)
      F(N4) = R(K9)
      IF(F(LHIGH).GT.F(N4)) GO TO 68
      DO 69 J=1, NX
      DO 69 I=1, N1
 69   X2(I, J) = 0.5 * (X2(I, J) + X2(LOW, J))
```

```
      DO 70 I=1, N1
      DO 71 J=1, NX
71    X(J)=X2(I, J)
      IN=I
      CALL SUMR
      SR(I)=SQRT(SEQL)
      IF(SR(I).LT.FDIFER) GO TO 72
      INF=I
      CALL FEASBL
      IF(FOLD.LT.1.0E-09) GO TO 80
72    CALL PROB(3)
70    F(I)=R(K9)
      GO TO 1000
68    DO 73 J=1, NX
73    X2(LHIGH, J)=X2(N4, J)
      SR(LHIGH)=SR(N4)
      F(LHIGH)=F(N4)
      GO TO 1000
81    WRITE(6, 760)  ICONT, FDIFER
      CALL WRITEX
      GO TO 9999
80    WRITE(6, 760)  ICONT, FDIFER
      CALL WRITEX
      WRITE(6, 762)
1     FORMAT(3I5, F10.5, E10.3)
2     FORMAT(8F10.5)
35    FORMAT(/, 40X, 48H* * * * * * * * * * * * * * * * * * * *)
106   FORMAT(1H1, //)
115   FORMAT(//, 41H THE STARTING VECTOR SELEC-
         TED BY USER IS)
116   FORMAT(8E16.6)
755   FORMAT(//, 35H THE COMPUTATION TIME IN SE-
```

```
      CONDS=E12.5)
756  FORMAT(//, 10X, 40H NUMBER OF INDEPENDENT
       VARIAB ES
     I5, /, 10X , 40H NUMBER OF EQUALITY CONSTR-
       AINTS I5,/, 10X, 40H NUMBER O F INEQUALETY
       CONSTRAINTS  I5, /, 10X, 40H SIZE OF INITIAL
     POLY HEDRON                E12.5, /, 10X, 40H THE
       DESIRED CONVERGENCE IS
       E12.5, /, 10X, 40H THE COMPUTATION TIME IN
       SECONDS
          E12.5)
757  FORMAT(//, 71H THE INITIAL X VECTOR DOES
       NOT SATISFY THE INITIAL T LERANCE CRITE
       RION)
758  FORMAT(/, 10X, 27H STAGE CALCULATION NUMB-
       ER=I5, 20X, 27H THE TOLER ANCE CRITERION=
       E14.6)
759  FORMAT(1H0, 3H* * * )
760  FORMAT(//, 39H TOTAL NUMBER OF STAGES CA-
       LCULATIONS=I5, 10X, 25H
       THE CONVERGENCE LIMIT=  E14.6)
761  FORMAT(//, 50X, 25H THESE ARE FINAL ANSW-
       ERS)
762  FORMAT(//, 50X, 29H THESE ARE NOT FINAL AN-
       SWERS    )
763  FORMAT(//, 10X, 40H THE INITIAL TOLERANCE
     CRITERION IS E12.5, /, 10X, 40H THE SUM OF V-
     IOLATED CONSTRAINTS      IS E12.5)
764  FORMAT(//, 70H THE VECTOR FOUND BY PROGR-
       AM WHICH SATISFIES THE INIT IAL TOLERA-
       NCE IS    )
765  FORMAT(/, 31H SUM OF VIOLATED CONS TRAI-
```

```
      NTS=E17.7)
9999 STOP
     END
     SUBROUTINE FEASBL
     * * * SUBROUTINE FEASBL MINIMZES THE SUM
C    OF THE SQUARE VALUES OF THE
C    VIOLATED CONSTRAINTS.IT IS CALLED EVERY
C    TIME THE COMBINED VALUE OF
C    THE VIOLATED CONSTRAINTS EXCEEDS THE VA
C        LUE OF THE TOLERANCE CRITER
C    ION FOR THE CURRENT STAGE
     DIMENSION X(50), X1(50, 50), X2(50, 50), R(100),
       SUM(50), F(50), SR(50)
     DIMENSION ROLD(100), R1(100), R2(100), R3(100),
       FLG(10), H(50)
     COMMON/A/NX, NC, NIC, STEP, ALFA, BETA, GA-
       MA, IN, INF, FDIFER, SEQL, K1, K2, 1K3, K4,
       K5, K6, K7, K8, K9, X, X1, X2, R, SUM, F, SR,RO-
        LD, SCALE, FOLD
     COMMON/B/LFEAS, L5, L6, L7, L8, L9, R1A, R2A,
       R3A
     ALFA=1.
     BETA=0.5
     GAMA=2.
     XNX=NX
     ICONT=0
     LCHEK=0
     ICHEK=0
  25 CALL START
     DO 3 I=1, K1
     DO 4 J=1, NX
   4 X(J)=X1(I, J)
```

```
      IN=I
      CALL SUMR
    3 CONTINUE
C      SELECT LARGEST VALUE OF SUM(I) IN SIMPLEX
   28 SUMH=SUM(1)
      INDEX=1
      DO 7 I=2, K1
      IF(SUM(I).LE.SUMH) GO TO 7
      SUMH=SUM(I)
      INDEX=I
    7 CONTINUE
C      SELECT MINIMUM VALUE OF SUM(I) IN SIMPLEX
      SUML=SUM(1)
      KOUNT=1
      DO 8 I=2, K1
      IF(SUML.LE.SUM(I)) GO TO 8
      SUML=SUM(I)
      KOUNT=I
    8 CONTINUE
C      FIND CENTROID OF POINTS WITH I DIFFERENT
C        THAN INDEX
      DO 9 J=1, NX
      SUM2=0
      DO 10 I=1, K1
   10 SUM2=SUM2+X1C(I, J)
      X1(K2, J)=1./XNX*(SUM2-X1(INDEX, J))
C      FIND REFLECTION OF HIGH POINT THROUGH CE-
C        NTROID
      X1(K3, J)=2.*X1(K2, J)-X1(INDEX, J)
    9 X(J)=X1(K3, J)
      IN=K3
      CALL SUMR
```

```
      IF(SUM(K3).LT.SUML) GO TO 11
C     SELECT SECOND LARGEST VALUE IN SIMPLEX
      IF(INDEX.EQ.1) GO TO 38
      SUMS=SUM(1)
      GO TO 39
   38 SUMS=SUM(2)
   39 DO 12 I=1, K1
      IF((INDEX-I).EQ.0) GO TO 12
      IF(SUM(I).LE.SUMS) GO TO 12
      SUMS=SUM(I)
   12 CONTINUE
        IF(SUM(K3). GT. SUMS) GO TO 13
      GO TO 14
C     FORM EXPANSION OF NEW MINIMUM IF REFLE-
C       CTION HAS PRODUCED ONE MIN.
   11 DO 15 J=1, NX
      X1(K4, J)=X1(K2, J)+2.*(X1(K3, J(-X1(K2, J))
   15 X(J)=X1(K4, J)
      IN=K4
      CALL SUMR
      IF(SUM(K4).LT.SUML) GO TO 16
      GO TO 14
   13 IF(SUM(K3).GT.SUMH) GO TO 17
      DO 18 J=1, NX
   18 X1(INDEX, J)=X1(K3, J)
   17 DO 19 J=1, NX
      X1((K4, J)=0.5*X1(INDEX, J)+0.5*X1(K2, J)
   19 X(3)=X1(K4, J)
      IN=K4
      CALL SUMR
      IF(SUMH.GT.SUM(K4(GO TO 6)
C     REDUCE SIMPLEX BY HALF IF REFLECTION HA-
```

```
         PPENS TO PRODUCE A LARGER
C        VALUE THEN THE MAXIMUM
         DO 20 J=1, NX
         DO 20 I=1, K1
   20    X1(I, J)=0.5*(X1(I, J)+X1(KOUNT, J))
         DO 29 I=1, K1
         DO 30 J=1, NX
   30    X(J)=X1(I, J)
         IN=I
         CALLSUMR
   29    CONTINUE
    5    SUML=SUM(1)
         KOUNT=1
         DO 23 I=2, K1
         IF(SUML.LT.SUM(I)) GO TO 23
         SUML=SUM(I)
         KOUNT=I
   23    CONTINUE
         SR(INF)=SQRT(SUM(KOUNT))
         DO 27 J=1, NX
   27    X(J)=X1(KOUNT, J)
         GO TO 26
    6    DO 31 J=1, NX
   31    X1(INDEX, J)=X1(K4, J)
         SUM(INDEX)=SUM(K4)
         GO TO 5
   16    DO 21 J=1, NX
         X1(INDEX, J)=X1(K4, J)
   21    X(J)=X1(INDEX, J)
         SUM(INDEX)=SUM(K4)
         SR(INF)=SQRT(SUM(K4))
         GO TO 26
```

```
14  DO 22 J=1, NX
    X1(INDEX, J)=X1(K3, J)
22  X(J)=X1(INDEX, J)
    SUM(INDEX)=SUM(K3)
    SR(INF)=SQRT(SUM(K3))
26  ICONT=ICONT+1
    DO 36 J=1, NX
36  X2(INF, J)=X(J)
    IF(ICONT.LT.2*K1) GO TO 50
     ICONT=0
     DO 24 J=1, NX
24  X(J)=X1(K2, J)
    IN=K2
    CALL SUMR
    DIFER=0.
    DO 57 I=1, K1
57  DIFER=DIFER+(SUM(I)-SUM(K2))**2
    DIFER=1./(K7*XNX)*SQRT(DIFER)
    IF(DIFER.GT.1.0E.-14) GO TO 50
C       IF FLEXIBLE SIMPLEX METHOD FAILED TO SAT-
C       ISFY THE CONSTRAINTS WITH-
C       IN  THE TOLERANCE CRITERION  FOR THE CUR-
C         RENT STAGE, THE SEARCH IS
C        PERTURBED FROM THE POSITLON WHERE THE X
C         VECTOR IS STUCK AND THEN
C       FEASBL IS REPEATED ONCE MORE FROM THE BE
C         GINNING
51  IN=K1
    STEP=20.*FDIFER
    CALLSUMR
    SR(INF)=SQRT(SEQL)
    DO 52 J=1, NX
```

```
52  X1(K1, J) =X(J)
    DO 53 J=1, NX
    FACTOR=1.
    X(J) =X1(K1, J) + FACTOR * STEP
    X1(L9, J) =X(J)
    IN=L9
    CALL SUMR
    X(J) =X1(K1, J) - FACTOR * STEP
    X1(L5, J) =X(J)
    IN=L5
    CALL SUMR
56  IF(SUM(L9).LT.SUM(K1)) GO TO 54
    IF(SUM(L5).SUM(K1)) GO TO 55
    GO TO 97
54  X1(L5, J) =X1(K1, J)
    SUM(L5) = SUM(K1)
    X1(K1, J) =X1((L9, J)
    SUM(K1) = SUM(L9)
    FACTOR=FACTOR+1.
    X(J) =X1(K1, J) + FACTOR * STEP
    IN=L9
    CALL SUMR
    GO TO 56
55  X1(L9, J) =X1(K1, J)
    SUM(L9) = SUM(K1)
    X1(K1, J) =X1(L5, J)
    SUM(K1) = SUM(L5)
    FACTOR=FACTOR+1.
    X(J) =X1(K1, J) - FACTOR * STEP
    IN =L5
    CALL SUMR
    GO TO 56
```

```
C       ONE DIMENSIONAL SEARCH BY GOLDEN SECTI-
C       ON ALONG EACH COORDINATE
      7 H(J)=X1(L9, J)-X1(L5, J)
        X1(L6, J)=X1(L5, J)+H(J)*R1A
        X(J)=X(L6, J)
        IN=L6
        CALL SUMR
        X1(L7, J)=X1(L5, J)+H(J)*R2A
        X(J)=X1(L7, J)
        IN=L7
        CALL SUMR
        IF(SUM(L6).GT.SUM(L7)) GO TO 68
        X1(L8, J)=X1(L5, J)+(1.-R3A)*H(J)
        X1(L5, J)=X1(L7, J)
        X(J)=X1(L8, J)
        IN=L8
        CALL SUMR
        IF(SUM(L8).GT.SUM(L6)) GO TO 76
        X1(L5, J)=X1(L6, J)
        SUM(L5)=SUM(L6)
        GO TO 75
     76 X1(L9, J)=X1(L8, J)
        SUM(L9)=SUM(L8)
        GO TO 75
     68 X1(L9, J)=X1(L6, J)
        X1(L8, J)=X1(L5, J)+R3A*H(J)
        X(J)=X1(L8, J)
        IN=L8
        CALL SUMR
        SUM(L9)=SUM(L6)
        IF(SUM(L7).GT.SUM(L8)) GO TO 71
        X1(L5, J)=X1(L8, J)
```

```
      SUM(L6) = SUM(L8)
      GO TO 75
 71   X1(L9, J) =X1(L7, J)
      SUM(L9) = SUM(L7)
 75   IF(ABS(X1(L9, J)-X1(L5, J)).GT.0.01*FDIFER)GO TO
        97
      X1(K1, J) =X1(L7, J)
      X(J) =X1(L7, J)
      SUM(K1) = SUM(L5)
      SR(INF) =   SQRT(SUM(K1))
      IF(SR(INF).LT.FDIFER) GO TO 760
 53   CONTINUE
      ICHEK = ICHEK + 1
      STEP = FDIFER
      IF(ICHEK.LE.2) GO TO 25
      FOLD = 1.0E-12
      WRITE(6, 853)
      WRITE(6, 850)
      WRITE(6, 851) (X(J), J=1, NX)
      WRITE(6, 852) FDIFER, SR(INF)
      GO TO 46
760   DO 761 J=1, NX
      X2(INF, J) =X1(K1, J)
761   X(J) =X1(K1, J)
 50   IF(SR(INF).GT.FDIFER) GO TO 28
C     MODIFIED LAGRANGE INTERPOLATION FOR TI-
C       GHTINEQUALITIES
      IF(SR(INF).GT.0) GO TO 35
      CALL PROB(3)
      FINT =R(K9)
      DO 139 J=1, NX
139   X(J) =X2(INF, J)
```

```
      CALL PROB(2)
      DO 40 J=K7, K8
40    R1(J)=R(J)
      DO 41 J=1, NX
41    X(J)=X1(KOUNT, J)
      CALL PROB(2)
      DO 42 J=K7, K8
42    R3(J)=R(J)
      DO 43 J=1, NX
      H(J)=X1(KOUNT, J)-X2(INF, J)
43    X(J)=X2(INF, J)+0.5*H(J)
      CALL PROB(2)
      FLG(1)=0.
      FLG(2)=0.
      FLG(3)=0.
      DO 44 J=K7, K8
      IF(R3(J).GE.0) GO TO 44
      FLG(1)=FLG(1)+R1(J)*R1(J)
      FLG(2)=FLG(2)+R(J)*R(J)
      FLGK(3)=FLG(3)+R3(J)*R3(J)
44    CONTINUE
      SR(INF)=SQRT(FLG(1))
      IF(SR(INF).LT.FDIFER) GO TO 35
      ALFA1=FLG(1)-2.*FLG(2)+FLG(3)
      BETA1=3.*FLG(1)-4.FLG(2)+FLG(3)
      RATIO=BETA1/(4.*ALFA1)
      DO 45 J=1, NX
45    X(J)=X2(INF, J)+H(J)*RATIO
      IN=INF
      CALL SUMR
      SR(INF)=SQRT(SEQL)
      IF(SR(INF).LT.FDIFER) GO TO 465
```

```
      DO 49 I=1, 20
      DO 48 J=1, NX
48    X(J)=X(J)-0.05*H(J)
      CALL SUMR
      SR(INF)=SQRT(SEQL)
      IF(SR(INF).LT.FDIFER) GO TO 465
49    CONTINUE
465   CALL PROB(3)
      IF(FINT.GT.R(K9)) GO TO 46
      SR(INF)=0.
      GO TO 35
46    DO 47 J=1 NX
47    X2(INF, J)=X(J)
35    CONTINUE
      DO 335 J=1, NX
335   X(J)=X2(INF, J)
850   FORMAT(//108H IT IS NOT POSSIBLE TO SATISFY
      THE VIOLATED CONSTRAIN2CHOOSE A NEW ST
      RTING VECTOR AND REPEAT SOLUTION AGAIN-
851   FORMAT(//, 63H THE VECTOR FOR WHICH THE
        CONSTRAINTS COULD NOT BE SA
      1TI FIED IS/, (8E16, 6))
852   FORMAT(//, 27H, THE TOLERANCE CRITERION=
       E14.6, 20X, 49H THE SQUARE
      1ROOT OF THE CONSTRAINTS SQUARED IS=E16.6)
853   FORMAT(//, 81H******SUBROUTINE FEASBL
      FAILS TO FIND A FEASIBLE POINT******)
      RETURN
      END
      SUBROUTINE WRITEX
      DIMENSION X(50), X1(50, 50), X2(50, 50), R(100),
```

```
      SUM (50), F(50), SR(50)
      DIMENSION ROLD(100)
      COMMON/A/NX, NC, NIC, STEP, ALFA, BETA,
     GAMA, IN, INF, FDIFER, SEQL, K1, K2, K3, K4,K5,
      K6, K7, K8, K9, X, X1, X2, R, SUM, F, SR, ROLD,
      SCALE, FOLD
      COMMON/B/LFEAS, L5, L6, L7, L8, L9, R1A, R2A,
      R3A
      CALL PRDB(3)
      WRITE (6, 1)R( K9)
    1 FORMAT(/, 28H OB JECTIVE FUNCTION VALUE =E
      17.7)
      WRITE(6, 2)(X(J), J=1, NX)
    2 FORMAT(/,29H THE INDEPENDENT VECTORS ARE
      /(6E17.7))
      IF(NC.EQ.0) GO TO 6
      CALL PROB(1)
      WRITE(6, 3)(R(J), J=1, NC)
    3 FORMAT(/,36H THE EQUALITY CONSTRAINT VAL-
      UE ARE /(6E17.7))
    6 IF(NIC.EQ.0)GO TO 5
      CALL PROB(2)
      WRITE(6. 4) (R(J), J=K7, K6)
    4 FORMAT(/,34H THE INEQUALITY CONSTRAINT V-
      ALUES/(6E17.7))
    5 RETURN
      END
      SUBROUTINE SUMR
C      * * THIS SUBROUTINE COMPUTES THE SUM OF
C     TH  SQUARE VALUES OF THE
C     VIOLATED CONSTRAINTS IN ORDER TO BE CO-
C     MPARED WITH THE TOLERANCE
```

```
C       CRITERION
        DIMENSION X(50), X1(50, 50), X2(50, 50), R(100),
          SUM(50), F(50), SR(50)
        DIMENSION ROLD(100)
        COMMON/A/NX, NC, NIC, STEP, ALFA,BETA,GA-
          MA,IN, INF, FDIFER, SEQL, K1, K2, K3,K4,K5,
          K6, K7, K8, K9, X, X1, X2, R, SUM, F,SR,ROLD,
          SCALE, FOLD
        COMMON/B/LFEAS, L5, L6, L7, L8, L9, R1A,R2A,
          R3A
        SUM(IN)=0.
        CALL PROB(2)
        SEQL=0.
        IF(NIC.EQ.0) GO TO 4
        DO 1 J=K7, K8
        IFR(J).GE.0.) GO TO 1
        SEQL=SEQL+R(J)*R(J)
      1 CONTINUE
      4 IF(NC.EQ.0) GO TO 3
        CALL PROB(1)
        DO 2 J=1, NC
      2 SEQL=SEQL+R(J)*R(J)
      3 SUM(IN)=SEQL
      5 RETURN
        END
        SUBROUTINE START
        DIMENSION X(50), X1(50, 50), X2(50, 50), R(100),
          SUM(50), F(50), SR(50)
        DIMENSION A(50, 50), ROLD(100)
        COMMON/A/NX, NC, NIC, STEP, ALFA, BETA,
        GAMA,IN,INF,FDIFER,SEQL,K1,K2,K3,K4,K5, K6,
        K7, K8, K9, X, X1, X2, R, SUM, F, SR, ROLD,
        SCALE, FOLD
        COMMON/8/LFEAS, L5, L6, L7, L8, L9, R1A, R2A,
          R3A
        VN=NX
```

```
      STEP1=STEP/(VN*SQRT(2.))*(SQRT(VN+1.)+VN
         -1.)
      STEP2=STEP/(VN*SQRT(2.))*(SQRT(VN+1.)-1.)
      DO 1 J=1, NX
    1 A(1, J)=0.
      DO 2 I=2, K1
      DO 4 J=1, NX
    4 A(I, J)=STEP2
      L=I-1
      A(I, L)=STEP1
    2 CONTINUE
      DO 3 I=1, K1
      DO3 J=1, NX
    3 X1(I, J)=X(J)+A(I, J)
      RETURN
      END
C     THIS IS A SUBROUTINE OF PROBLEM
      SUBROUTINE PROB (IND)
      DIMENSION R(5), X(2)
      COMMON/A/NX, NC, NIC, STEP, ALFA, BETA,
         GAMA, IN, INF, FOIFER, SEQL, K1, K2,
      1K3, K4, K5 ,K6, K7, K8, K9, X1, X2, R, SUM, F,
         SR, ROLD, SCALE, FOLD
C     EQUALITY CONSTRAINTS
      GO TO (1, 2, 3), INQ
    1 R(1)=25-X(1)**2-X(2)**2
      GO TO 5
C     INEQUALITY CONSTRAINTS
    2 R(2)=10*X(1)-X(1)**2+10*X(2)-X(2)**2-34
      R(3)=X(1)
      R(4)=X(2)
      GO TO 5
C     OBJECTIVE FUNCTION
    3 R(5)=4*X(1)-X(2)**2-12
    5 RETURN
      END
```

# 第四章　震源断层面解的数值方法

地震发生时，震源运动的一个重要结果是辐射地震波。地震纵波到达台站第一个振辐的方向(初动方向)，在地面分布呈现一种特殊的图象，这种图象反映了震源的运动过程。

目前，对于震源机制的研究多以断层说为基础。所要确定的震源参数包括断层面的走向、倾向、错动力的方向、断层面的尺度、地震发生时震源处主应力方向及应力降等。

利用$P$波初动求断层面解是目前国内外常用的方法。因为这种方法对资料的要求不高(仅需要读出初动的符号)，求解方法简单，因此从六十年代起就有人研究用计算机求断层面解，以提高精度和效率。在六十年代末，国外已实现了计算机处理。我国在七十年代初也根据有关文献作了适当改进，在计算机上处理了大量实际资料。下面讨论基本原理及数值计算方法。

## §4-1　$P$波初动解方法

### (一) 基 本 概 念

$P$ 波初动符号在地面上分布的图象，与震源运动过程中断裂的几何产状及力的作用形式有直接关系。震源运动过程可以用一个力学模式来表示。

最简单的是单力偶模式。如图4-1-1所示，在 $XOY$ 平面内，用沿 $X$轴作用的大小相等、方向相反的一对力来模拟震源运动。$XOZ$ 相当于地震发生时产生的断层面。根据理论计算，在 $XOZ$ 面和$YOZ$ 面相互正交所分成的四个象限区域内，属于力的正方

向的两个区域，辐射初动为压缩（+号）的纵波；属于力的反方向的两个区域，辐射初动为膨胀（－号）的纵波。显然，*XOZ*面和*YOZ*面成了纵波和两个节面。它们扩展后与地面相交，就造成了地面上纵波初动符号分布上正负相间的四象限，以及分隔这些象限的两条相交的节线。地面上的垂直向地震仪，在压缩纵波相对应的象限内，将观测到*P*波初动向上（+号），在膨胀纵波相对应的象限内，将观测到*P*波初动向下（－号）。

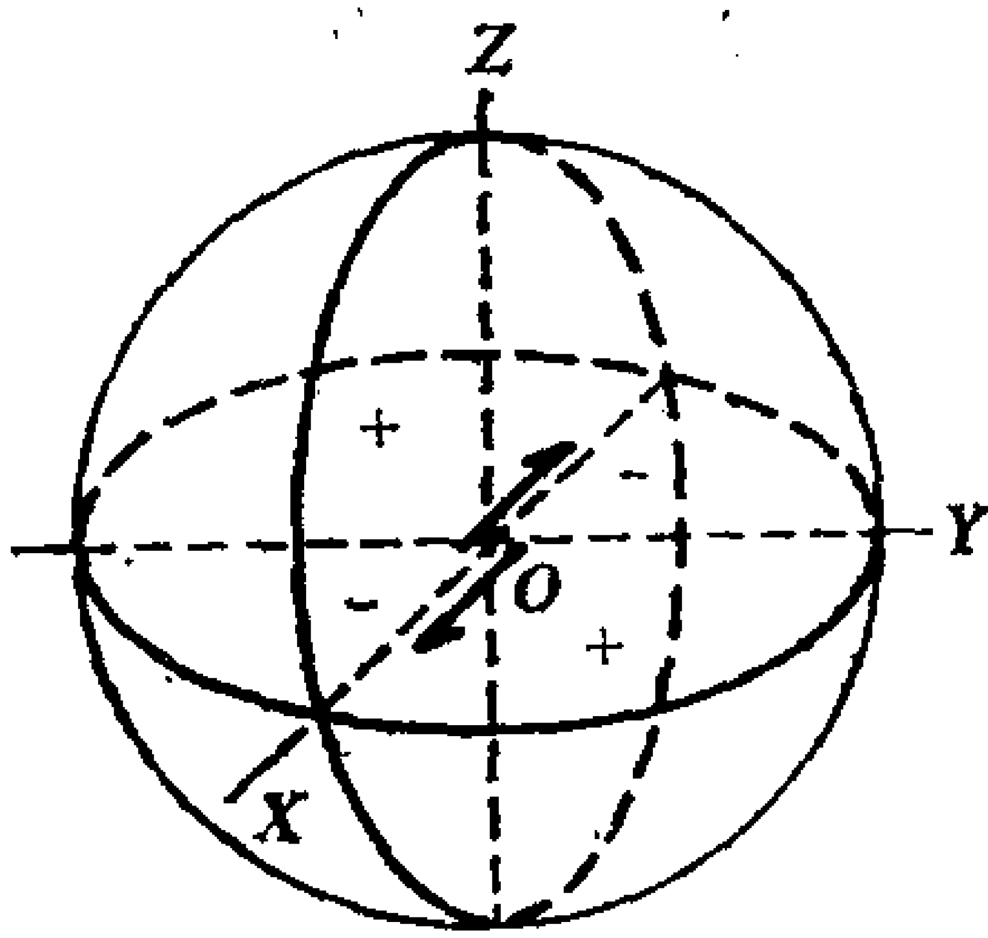


图 4-1-1

单力偶源所产生的*P*波辐射图象与地震发生时在地面的*P*波初动符号分布是一致的。但是，能够解释*P*波初动符号分布的还有一种双力偶模式。如图 4-1-2(a) 所示。在*XOY*面内，沿*X*轴和沿*Y*轴同时有大小相等方向相反的两对力的作用。可以证明它们完全等效于图4-1-2(b)中的主压力及主张力。

显然，主压力*P*与主张力*T*各与X轴和Y轴夹角45°，断层面可以是*XOZ*面，也可以是*YOZ*面。

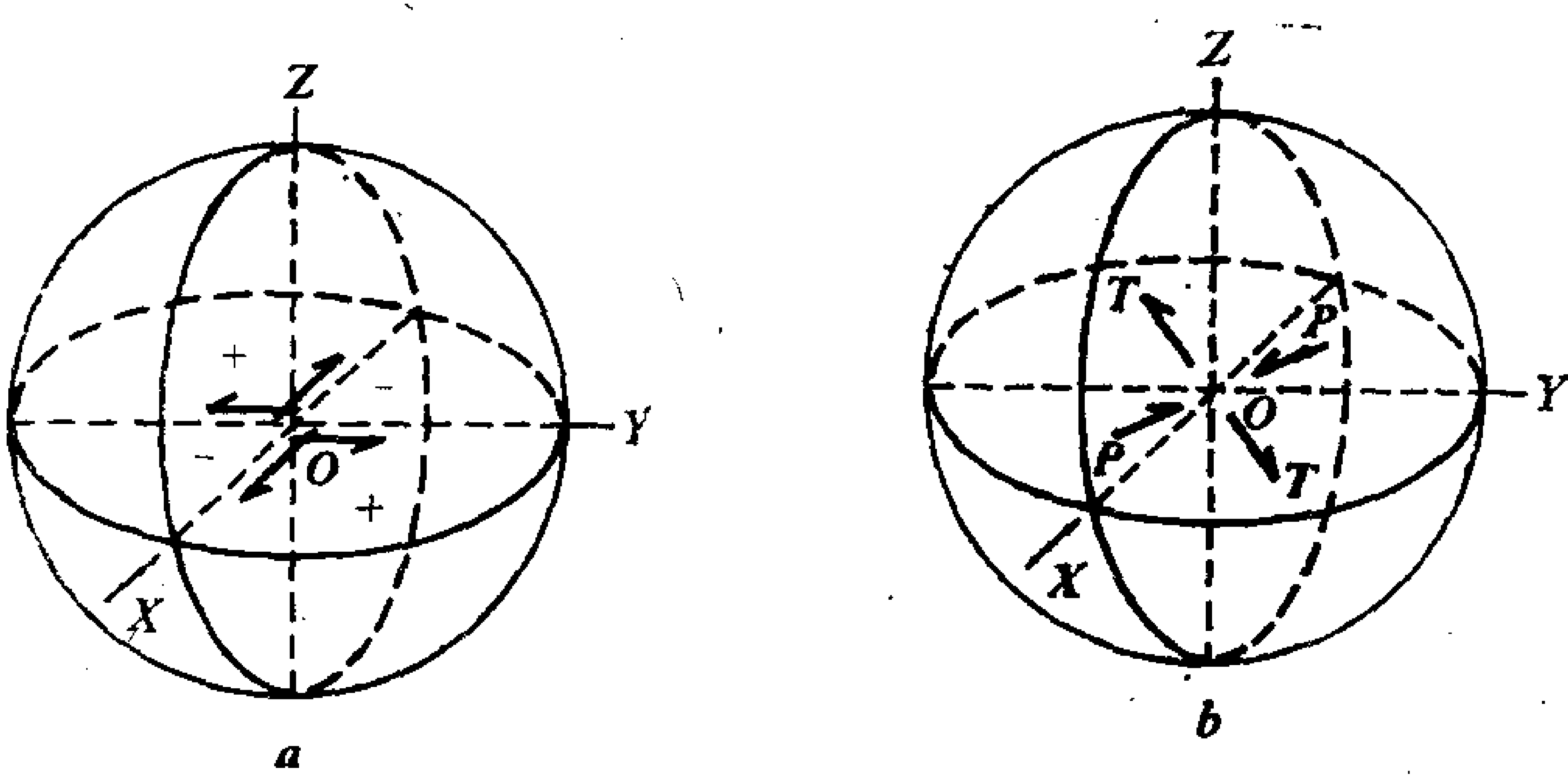


图 4-1-2

所以，$P$ 波初动解无法确定那个节面为断层面，只好认为两个节面都是可能的断层面。对于单力偶源，即求两个可能的断层面和两个可能力轴 $X$ 和 $Y$ 的空间位值；双力偶源则求两个可能的断层面，两个力轴 $X$ 和 $Y$，以及主压力轴$P$和主张力轴$T$的空间位置。

## （二）震源球面和乌尔夫网

假定直达纵波沿射线不改变初动符号。然而，由于纵波速度随深度增加引起射线弯曲，有可能使射线离源时所在的象限区域与射线出射地面时所对应的象限区域完全不同。如图 4-1-3 所示。

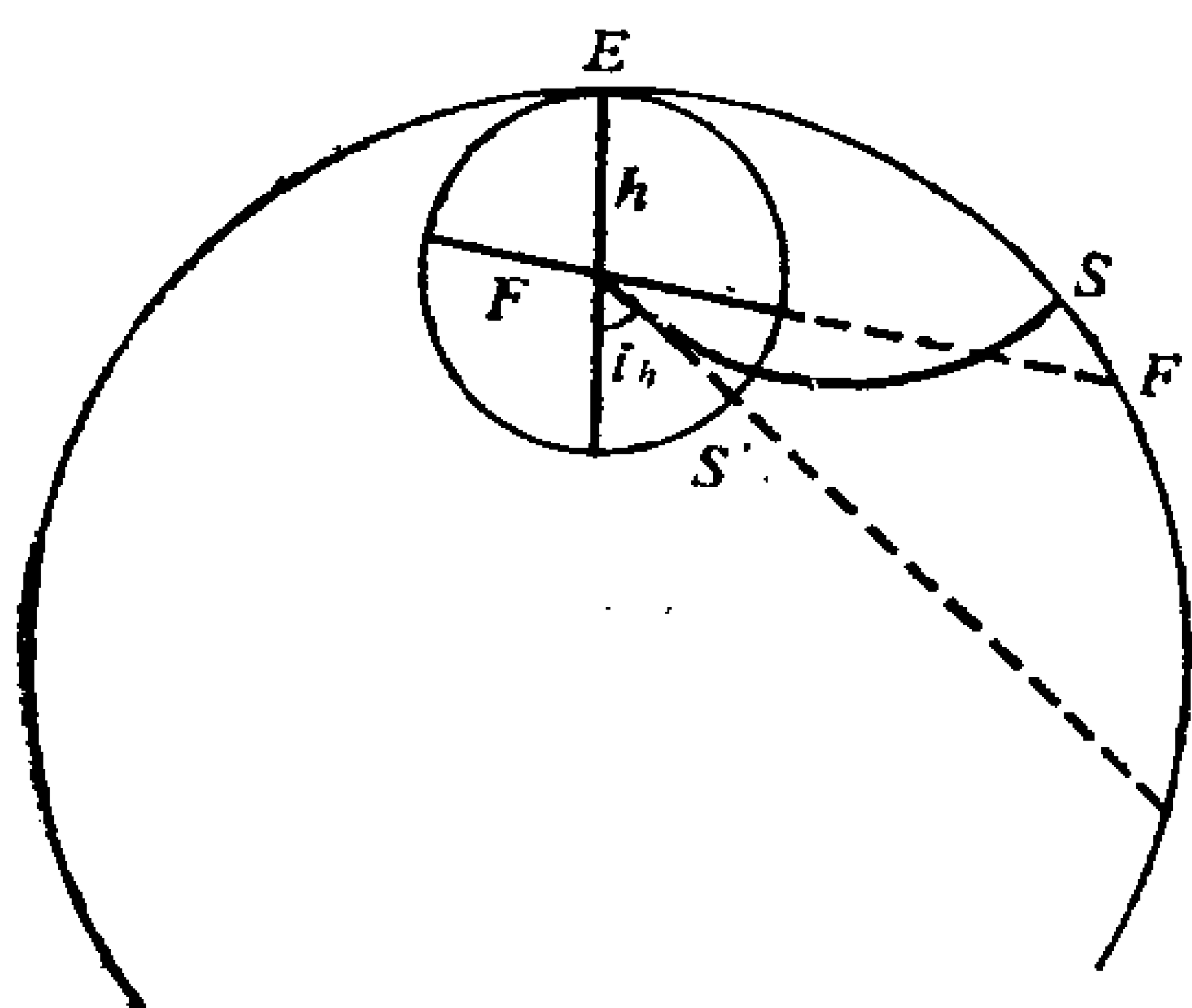


图 4-1-3

为了消除射线弯曲所造成的这种影响，必须将台站位置$S$移到地震射线在震源处的切线和一个以震源为球心的球面的交点$S'$上去。这个球面称为震源球面。它的半径可以取得任意小以便球内介质可视为均匀，射线弯曲可以忽略。震源处地震射线的切线方向与垂线方向夹角称为离源角，以$i_h$表示。对于各种纵波，都规定它取锐角。因此，直达$\overline{P}$波取射线向上发射方向与向上垂线的夹角；而$P$波取射线向下发射方向与向下垂线的夹角。

这样，观测台站在震源球面上的相应位置就由台站相对于震中的方位角和射线离源角唯一确定了。此外，在震源的两个相互正交的节平面就将震源球等分为四个象限区域，在球面上就为相互正交的两个大圆圈所等分的四块面积。

为了求解方便，须将震源球面上的点投影到平面上去。我们

采用赤极投影。如图 4-1-4 所示，这是震源球面在台站$S$方位上的剖面图。图中$O$为震源，$BB'$为投影平面，$A$为投射点。$S_1'$为台站$S$震源球面上的位置，则$S_1''$为其投影。对于射线向下发射的台站，其在震源球面上的位置出现在下半球上。考虑到节面所分成的四个象限区域中相对的两个区域辐射的$P$波初动符号相同，所以可以采用它的对蹠点投影。

在图4-1-4中，$S_2'$为射线向下发射的台站所相当的位置，$S_2'''$为其对蹠点，则$S_2''$为其投影。很容易看出，台站的方位线都投影成为自中心向外的辐射线。

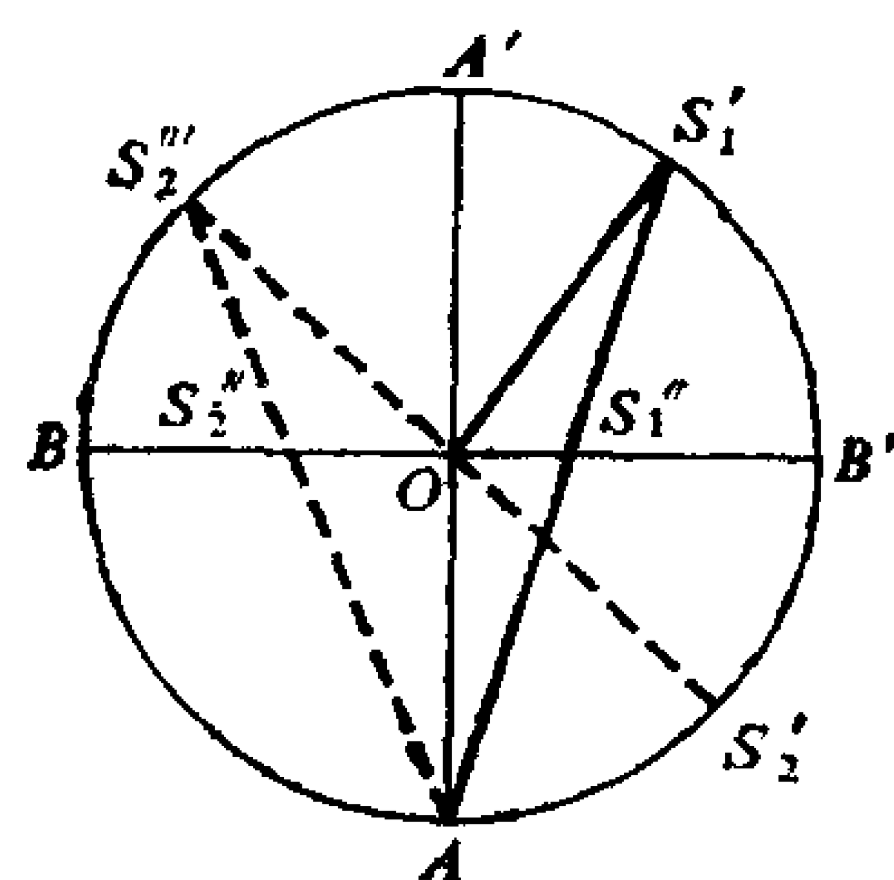


图 4-1-4

而震源球面上两条正交的大圆圈，投影后就成为正交的大圆弧。如图4-1-5示，$A$，$B$二圆在平面上的投影分别为$A'$，$B'$二弧。$A'$与$B'$是正交的。

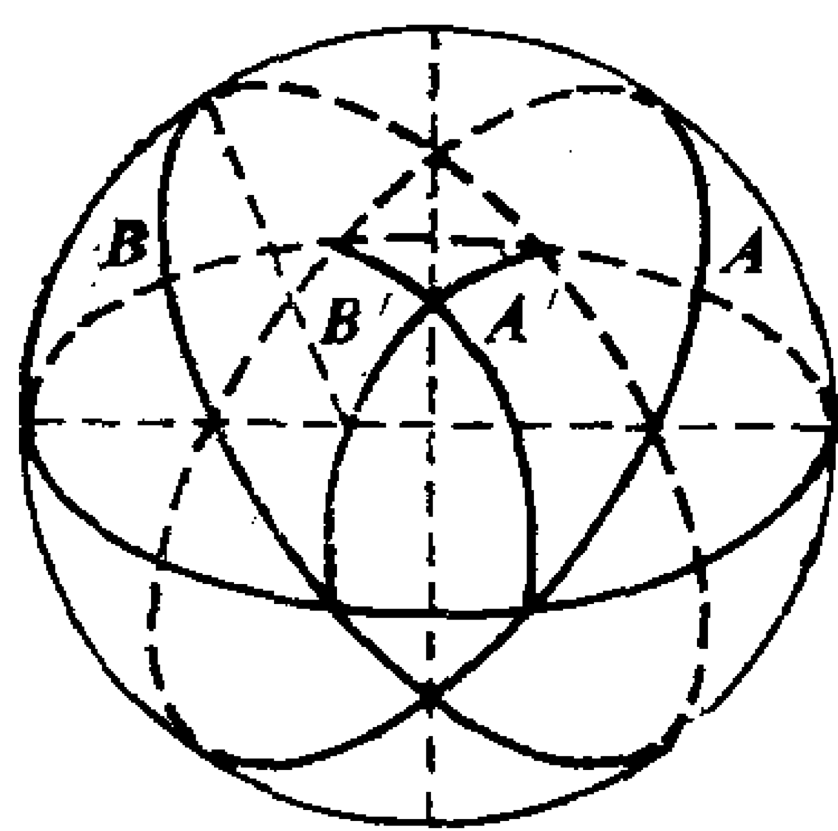


图4-1-5

利用这种投影方法可直接使用现成的乌尔夫网。我们选取图网中心为震源的投影，并规定网边某一点为正北，那么台站的方位角可由正北顺时针方向在网边的度数上求得。在台站的方位线上，根据射线为向上发射或向下发射，自网心向台站方向或背台站方向找到与离源角度数对应的点，就得到了台站位置的投影点$S_1''$或$S_2''$。

## （三）震源运动的参数

在乌尔夫网上找到每个台站的位置，在每个位置上标上各自的初动符号，再划两条大圆弧分隔它们为正负相间的区域。这两

条大圆弧就是$P$波的两条节线。因为节线是相互正交的，所以此二大圆弧也必须正交。正交的标志是一大圆弧的极点必落在另一大圆弧上（大圆弧的极点为大圆弧中垂线上与垂足相距90°的点）。如此得出的结果如图4-1-6所示。对比图4-1-2(a)，可以看出，$AA'$平面相当于$XOZ$平面。$BB'$平面相当于$YOZ$平面，$X_A$为$OX$轴出地点，与沿$OX$轴方向作用力出地点一致。$Y_B$为$OY$轴出地点，与沿$OY$轴方向作用力出地点一致。

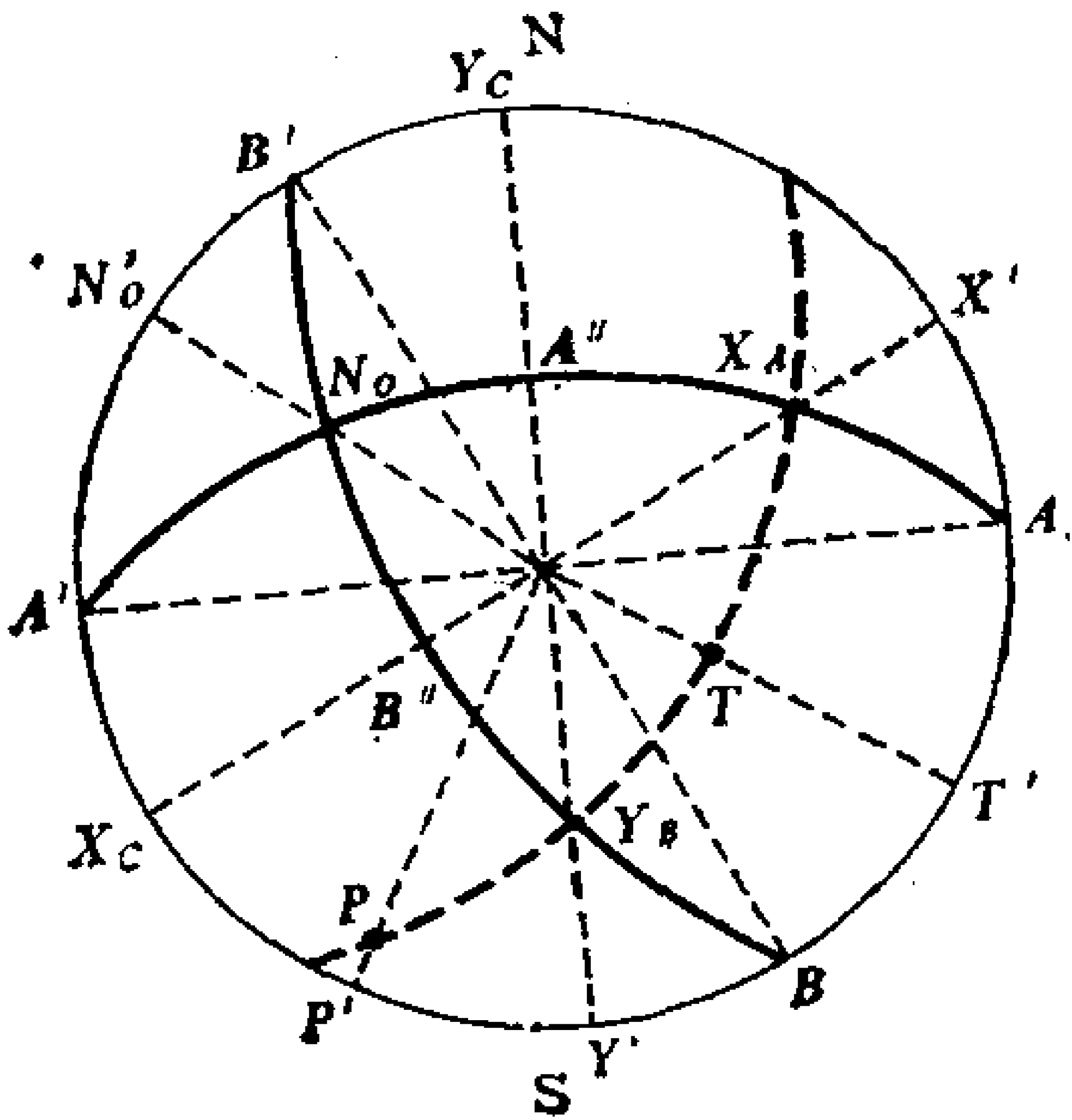


图 4-1-6

如果沿$X_A$与$Y_B$所在的大圆弧线上找出与$X_A$和$Y_B$相差弧距为45°的点$T$和$P$，对比图4-1-2(b)可知，$T$相当于$XOY$平面内主张力轴出地点，$P$相当于主压力轴出地点。$AA'$与$BB'$的交点$N_O$，相当于$OZ$轴。

由图4-1-6，可将所有参数列表如下：

| | 走向 | 倾向 | 倾角 |
|---|---|---|---|
| 节平面$A$ | $\widehat{NA}$ | $\widehat{NY'}$ | $Y_CA'$ |
| 节平面$B$ | $\widehat{NB}$ | $\widehat{NX'}$ | $X_CB'$ |
| | 走向 | | 仰角 |
| $X_A$轴 | $\widehat{NX'}$ | | $X'X_A$ |
| $Y_B$轴 | $\widehat{NY'}$ | | $Y'Y_B$ |
| $N_O$轴 | $\widehat{NN_O'}$ | | $N_O'N_O$ |
| $P$轴 | $\widehat{NP'}$ | | $P'P$ |
| $T$轴 | $\widehat{NT'}$ | | $T'T$ |

# § 4-2 $P$波初动解的数值计算

六十年代初，Knopoff(1961)提出应用数理统计原理求震源运动参数的方法。经过改进以后，可用作计算机处理(Kasahara，1963；Hodgson，1965)。以后，Wickens(1967)及Stevens(1968)又将方法进一步发展。这里给出的计算方法，主要来自Wickens等人的文章。

用数值方法求$P$波初动解，其关键问题是坐标轴的选取及概率函数的计算。现分述如下。

## (一) 坐标系的选择

取固定的参考坐标系为$X$，$Y$，$Z$。原点在震源$O$点，$X$轴向南，$Y$轴向东，$Z$轴向上。设$P$波两个节平面分别为$A$及$C$，彼此正交。因此，这两个节面的法线$A$，$C$与两节面的交线$B$(称零矢量)在震源$O$点组成一个正交系。我们称$A$，$B$，$C$为震源坐标系，它确定了节面的空间位置及各力轴的方向。这两组坐标系的关系如图 4-2-1 所示。

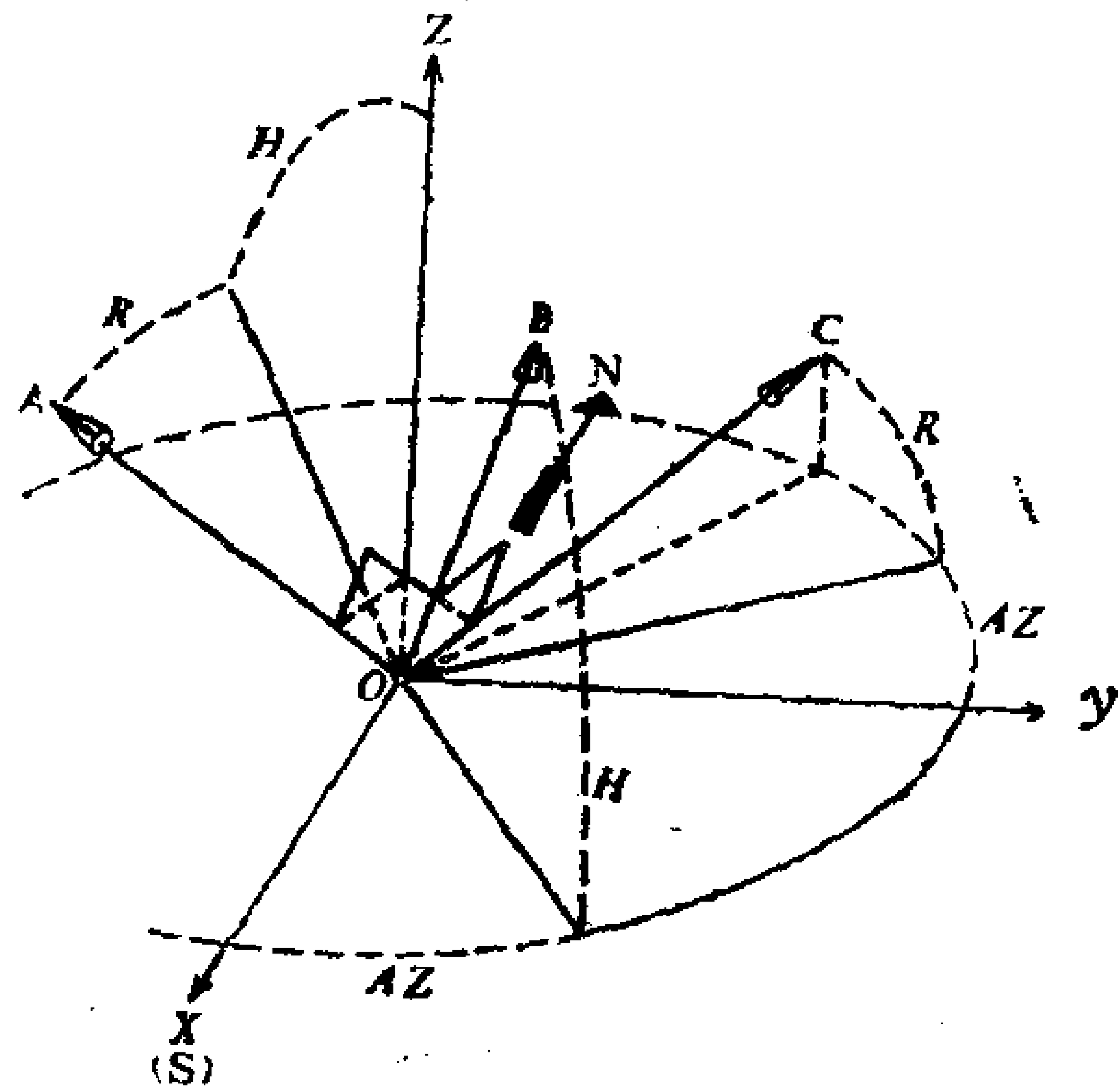



图 4-2-1

由图 4-2-1 可见，震源坐标$A$，$B$，$C$相对于固定坐标$X$，$Y$，$Z$的位置由三个角度，即$AZ$(0—360°)，$R$(0—90°)及$H$(0—90°)所决定。矢量$A$，$B$、$C$的方向余弦用下列式子表达：

$$A_X = -\sin H\cos AZ\cos R + \sin AZ\sin R,$$

$$A_Y = -\sin H \sin AZ \cos R - \cos AZ \sin R, \qquad (4\text{-}2\cdot1)$$
$$A_Z = \cos H \cos R.$$
$$B_X = \cos H \cos AZ,$$
$$B_Y = \cos H \sin AZ, \qquad (4\text{-}2\cdot2)$$
$$B_Z = \sin H.$$
$$C_X = -\sin H \cos AZ \sin R - \sin AZ \cos R,$$
$$C_Y = -\sin H \sin AZ \sin R + \cos AZ \cos R, \qquad (4\text{-}2\cdot3$$
$$C_Z = \sin R \cos H.$$

理论研究表明，地震振幅在震源球面上强度与所在位置有关。设 $\theta$ 为观察点矢径与矢量 $C$ 之间的夹角，$\phi$ 为观察点矢径在矢量 $A$、$B$ 组成的平面内的投影与矢量 $A$ 之间的夹角。$\alpha$ 为与震源强度成正比的系数。则在震源球面上第 $i$ 个台站的理论振幅(或理论信号水平)为

$$U_{Pi} = \alpha \sin 2\theta_i \cos\phi_i, \qquad (4\text{-}2\cdot4)$$

或者写为

$$U_{Pi} = 2\alpha \sin\theta_i \cos\theta_i \cos\phi_i. \qquad (4\text{-}2\cdot5)$$

因为我们采用理论相对振幅，所以将系数 $2\alpha$ 略去，并且使 $A$、$B$、$C$ 和射线方向 $p_i$ 都取单位矢量，这样

$$U_{Pi} = \sin\theta_i \cos\phi_i \cos\theta_i, \qquad (4\text{-}2\cdot6)$$

而

$$\sin\theta_i \cos\phi_i = p_i \cdot A,$$
$$\cos\theta_i = p_i \cdot C.$$

所以，第 $i$ 个台站的相对理论振幅成为

$$U_{Pi} = (p_i \cdot A)(p_i \cdot C). \qquad (4\text{-}2\cdot7)$$

射线矢量 $p_i$ 的方向余弦可以用离源角 $i_h$ 及台站方位角 $Z$ 来表示：

$$P_{Xi} = -\cos Z \sin i_h,$$
$$P_{Yi} = \sin Z \sin i_h, \qquad (4\text{-}2\cdot8)$$
$$P_{Zi} = -\cos i_h.$$

## （二）概 率 函 数

一个可靠的断层面解答要求台站所提供的初动符号能用两个节面分开，首先需要选择一个概率函数来描述这种分布的好坏。根据数理统计原理，一个台站提供的初动符号正确与否，既取决于台站记录的信噪比（这与仪器性能、台站干扰背景等因素有关），也取决于台站相对于震源的位置。

由(4-2·7)式可以看出，在节面$A$和$C$上（分别为与矢量$A$及$C$垂直的方向），理论相对振幅为零，而在两等分角面上达到最大值。所以台站所提供的初动符号的可靠性，与其距节面的距离有很密切的关系。因为，靠近节面的台站初动振幅小，测不准的概率大，而远离节面的台站记到的初动振幅大，测不准的概率小。因此，概率函数可以用信噪比$U_{P_i}/|N_i|$来描述。其中，$U_P^i$为第$i$台的理论振幅，$N_i$是噪音水平。

Knopoff(1961)讨论了节平面的“最佳拟合”问题，他提出用下列概率函数表示台站观测与理论计算符合的衡量准则

$$\frac{1}{2}[1+\mathrm{erf}(U_{P_i}/|N_i|)\mathrm{sign}U_{P_i}\mathrm{sign}R_i]. \qquad (4\text{-}2\cdot9)$$

式中 erf 为误差函数，其定义为

$$\mathrm{erf}(U_{P_i}/|N_i|)=\frac{2}{\sqrt{\pi}}\int_0^{U_{P_i}/|N_i|}e^{-t^2}dt. \qquad (4\text{-}2\cdot10)$$

而 sign 是理论振幅的符号函数；sign$R_i$是观测振幅的符号函数。符号函数的定义为

$$\mathrm{sign}\,X=\begin{cases}1 & X>0,\\ 0 & X=0,\\ -1 & X<0.\end{cases} \qquad (4\text{-}2\cdot11)$$

台站观测的初动符号与理论计算符号一致时称为符合，相反时称为不符合。从(4-2·9)式可以看出，台站符合时概率大，不

符合时概率小。$M$ 个台站同时符合的概率为

$$\prod_{i=1}^{M}\frac{1}{2}[1+\mathrm{erf}(U_{P_i}/|N_i|)\mathrm{sign}U_{P_i}\mathrm{sign}R_i]. \qquad (4\text{-}2\cdot12)$$

当所选择的节面位置使这个函数值取最大值时，所对应的节面位置就是最佳解。

为了计算的方便，对(4-2·12)式取对数，并令

$$\Phi=\sum_{i=1}^{M}\log\frac{1}{2}[1+\mathrm{erf}(U_{P_i}/|N_i|)\mathrm{sign}U_{P_i}\mathrm{sign}R_i]. \qquad (4\text{-}2\cdot13)$$

取不同节面的位置，则有不同的$\Phi$值。$\Phi$的最大值意味着在$M$个台站中符合的个数已达到最高数。

在公式(4-2.13)中设

$$W_{P_i}=\frac{1}{2}\mathrm{erf}(U_{P_i}/|N_i|) \qquad (4\text{-}2\cdot14)$$

为权函数。(4-2·14)的物理意义比较清楚，但计算时要作积分。这在大量计算时是不方便的。Wickens经过大量试算找到一个近似函数 $W_{P_i}$ 来代替(4-2·14)式作为权函数

$$W_{P_i}=|2U_{P_i}(1-|U_{P_i}|)+K\mathrm{sign}U_{P_i}|. \qquad (4\text{-}2\cdot15)$$

通过实验证明，(4-2·14)式与(4-2·15)式很接近。式中 $K$ 为待定系数，从经验中得出用0.3效果较好。综合上面的讨论，可得出最佳节面位置的函数关系为下列形式：

$$S=\frac{\left|\sum_{i=1}^{M}W_{P_i}\mathrm{sign}U_{P_i}\mathrm{sign}R_i\pm D\right|}{\sum_{i=1}^{M}|W_{P_i}|}. \qquad (4\text{-}2\cdot16)$$

取绝对值的意义是因为断层面只能确定两个相互垂直的节面，所以$S$ 取最大的正值和最大的负值是等价的。其中$D$ 是修正项，

按下式取值

$$D = 0.3[\min(N, P)]. \qquad (4\text{-}2\cdot17)$$

式中$N$和$P$是在第$K$和第$K+1$两个相邻节面位置时观测的正初动符号和负初动符号的个数，$D$是取两者中小的一个乘上0.3。$D$的正负号选取在于使$S$值增大。

显然，一组$AZ$、$H$、$R$的数值可以用(4-2·16)式算出一个$S$值。而这一组$AZ$、$H$、$R$的数值又代表一组断层面解。计算的$S$值愈大，表明理论与观测符合得愈好，即求得的节面解与待求的断层面解愈接近。

由于$AZ$，$R$，$H$的可变范围很大，分别为0°—360°，0°—90°，0°—90°。要直接解出$S$的最大值有一定困难，所以采用试探法来计算。计算可分为两个步骤。

第一步是粗选。$AZ$每隔30°取一值，$R$及$H$每隔15°取一值。这样所形成的试验组合为432个，将每种组合代入(4-2·16)式，共求出432个$S$值来，比较这些$S$值，从中选出最大的$N$个。

第二步是围绕这$N$个最大的$S$值进行细选。细选是浮动的，即一方面是寻求$S$的峰值，同时又起到加密间隔的作用，使原来差别较大的解收敛于一个峰值附近。具体算法可用类似于单纯形法那样作细分寻查。对$N$个粗选的$S$值分别作浮动细选，得到$N$个精选的$S$值。最后从中挑选出5个最大的$S$值算出相应的断层面参数并画出图。

## §4-3 各有关参数的计算公式

上一节曾指出，射线矢量$p_i$的方向余弦可以用台站方位角z及离源角$i_h$来表示。所以必须求下列参数：

### （一）台站方位角及震中距

台站震中距$\Delta$及方位角Z分别用下列球面公式进行计算：

$$\cos\Delta = \cos\phi_e\cos\phi_s + \sin\phi_e\sin\phi_s\cos(\lambda_e - \lambda_s), \qquad (4\text{-}3\cdot1)$$

$$\cos Z = (\cos\phi_s - \cos\phi_e\cos\Delta)/\sin\Delta\sin\phi_e,$$

$$\sin Z = -\sin\phi_s\sin(\lambda_e - \lambda_s)/\sin\Delta. \qquad (4\text{-}3\cdot2)$$

式中 $\lambda_e$：震中经度；$\lambda_s$：台站经度；$\phi_e$：震中的地心余纬度；

$\phi_s$：台站的地心余纬度；$\Delta$：震中距；$Z$：台站方位角。

地心纬度$\psi$与地理纬度$\phi'$的关系如下：

$$\operatorname{tg}\psi = (1-f)^2\operatorname{tg}\phi'.$$

$f$为地球的扁率

$$f = \frac{1}{298.25}.$$

地心余纬度的定义是

$$\phi = \begin{cases} 90° - \psi, (北纬); \\ 90° + \psi, (南纬). \end{cases}$$

## （二）离　源　角

我国境内绝大部分地震都发生在地壳中，对于震中距$\Delta < 2°$时，用直达波($\overline{P}$)公式计算离源角，

$$i_h = \operatorname{tg}^{-1}\frac{\Delta}{h}. \qquad (4\text{-}3\cdot3)$$

对于$P^*$及$P_n$波，其离源角取决于地壳模型。我国地壳一般采取双层模型，东部地区$h_1 = 15$ km，$h_2 = 20$ km；西部地区取$h_1 = 20$ km，$h_2 = 30$km。不论东部或西部，$V_1 = 6.0$km/s，$V_2 = 6.75$ km/s，地幔顶部的纵波速度为8.049km/s。

因此，对于$P^*$波，离源角按下式计算，

$$i_h = \sin^{-1}\left(\frac{6.0}{6.75}\right) \simeq 62.7° . \qquad (4\text{-}3\cdot4)$$

对于$P_n$波，当震源深度位于上层地壳时，

$$i_h = \sin^{-1}\left(\frac{6.0}{8.049}\right) \simeq 48.2°. \tag{4-3·5}$$

当震源深度位于下层地壳时(即h≥15或20公里)

$$i_h = \sin^{-1}\left(\frac{6.75}{8.049}\right) \simeq 57° . \tag{4-3·6}$$

对于 P波，由于地震台离震中较远，射线在地球内部是弯曲的，其离源角 $i_h$ 的计算方法如下：

当震源接近地面时，P 波射线是对称的，其离源角 $i_h$ 和射线到达地震台的入射角$i$近乎相等。P 波在地表的入射角为

$$\sin i_h = \frac{V_0}{\bar{V}}. \tag{4-3·7}$$

式中 $V_0$ 为地面附近波传播的真速度，$\bar{V}$ 为在地震台附近与震中距有关的P 波沿地面的视速度。$\bar{V}$ 是震中距$\Delta$ 的函数，故离源角也随震中距而变化。

当震源在地壳深处或地幔中时，P 波射线是不对称的。即离源角 $i_h$ 不等于入射角 $i$ 。其计算公式为

$$\sin i_h = \frac{h}{R-h} \frac{V_h}{V_0} \sin i. \tag{4-3·8}$$

式中$R$ 为地球半径，$V_h$ 为震源处P 波速度，$V_0$ 为地表 P 波速度，$h$ 为震源深度。入射角 $i$ 是震中距$\Delta$ 的函数。

因此，对于不同震源深度的地震，可以按震中距$\Delta$作出离源角表供查用。

## （三）震源运动参数公式

按(4-2·1)—(4-2·3)式给出的方向余弦，即可根据以下公式计算震源运动参数：

（1）A 节面的倾向方位

$$E_A = \text{arctg}\left(\frac{-\sin H\cos AZ\sin R - \sin AZ\cos R}{-\sin H\sin AZ\sin R + \cos AZ\cos R}\right)$$

$$= \text{arctg}\frac{C_x}{C_y}. \tag{4-3·9}$$

若$C_y<0$，则$E_A+270°$，否则$E_A+90°$。

（2） $C$节面的倾向方位

$$E_C = \text{arctg}\left(\frac{-\sin H\cos AZ\cos R + \sin AZ\sin R}{-\sin H\sin AZ\cos R - \cos AZ\sin R}\right)$$

$$= \text{arctg}\frac{A_x}{A_y}. \tag{4-3·10}$$

若$A_y<0$，则$E_C+270°$，否则$E_C+90°$。

（3） $A$节面的倾角

$$D_A = \text{arctg}\frac{[1-(\sin R\cos H)^2]^{1/2}}{\sin R\ \cos H}$$

$$= \text{arctg}\frac{(1-C_z^2)^{1/2}}{C_z}. \tag{4-3·11}$$

（4） $C$ 节面的倾角

$$D_C = \text{arctg}\frac{[1-(\cos H\cos R)^2]^{1/2}}{\cos H\ \cos R}$$

$$= \text{arctg}\frac{(1-A_z^2)^{1/2}}{A_z}. \tag{4-3·12}$$

（5） $P$轴的方位

若 $A_x \cdot C_x<0$或$A_y \cdot C_y<0$，则

$$E_P = \text{arctg}\frac{\sin H\cos AZ(\sin R-\cos R)+\sin AZ(\sin R+\cos R)}{\sin H\cos AZ(\sin R-\cos R)-\cos AZ(\sin R+\cos R)}$$

$$= \text{arctg}\frac{A_x-C_x}{A_y-C_y}. \tag{4-3·13}$$

若$A_y-C_y<0$，则$E_P+90°$，否则$E_P+270°$。

（6） $P$轴和水平面的夹角

$$D_P = 90^\circ - \text{arctg}\frac{\sqrt{2}\left[1-\frac{1}{2}(\cos H\cos R - \sin R\cos H)^2\right]^{1/2}}{\cos H(\cos R - \sin R)}$$

$$= 90^\circ - \text{arctg}\frac{\sqrt{2}\left[1-\frac{1}{2}(A_z - C_z)^2\right]^{1/2}}{A_z - C_z}. \qquad (4\text{-}3\cdot14)$$

(7) $B$ 轴的方位

$$E_B = \text{arctg}(\text{ctg}A_z) = \text{arctg}\frac{B_x}{B_y}. \qquad (4\text{-}3\cdot15)$$

若 $B_y<0$，则$E_B+90^\circ$，否则$E_B+270^\circ$。

(8) $B$ 轴和水平面的夹角

$$D_B = 90^\circ - \text{arctg}(\text{ctg}H)$$

$$= 90^\circ - \text{arctg}\frac{(1-B_z^2)^{1/2}}{B_z}. \qquad (4\text{-}3\cdot16)$$

(9) $T$ 轴的方位

若$A_x\cdot C_x<0$或$A_y\cdot C_y<0$则

$$E_T = \text{arctg}\frac{-\sin H\cos AZ(\cos R + \sin R) + \sin AZ(\sin R - \cos R)}{-\sin H\sin AZ(\cos R + \sin R) - \cos AZ(\sin R - \cos R)}$$

$$= \text{arctg}\frac{A_x + C_x}{A_y + C_y}. \qquad (4\text{-}3\cdot17)$$

若$A_y+C_y<0$，则$E_T+90^\circ$，否则$E_T+270^\circ$。

(10) $T$ 轴和水平面的夹角

$$D_T = 90^\circ - \text{arctg}\frac{\sqrt{2}\left[1-\frac{1}{2}(\cos R\cos H + \sin R\cos H)^2\right]^{1/2}}{\cos H(\cos R + \sin R)}$$

$$= 90^\circ - \text{arctg}\frac{\sqrt{2}\left[1-\frac{1}{2}(A_z + C_z)^2\right]^{1/2}}{A_z + C_z}. \qquad (4\text{-}3\cdot18)$$

若$A_y\cdot C_y>0$或$A_x\cdot C_x>0$，则$P$轴与$T$轴交换位置。

## §4-4 P波初动断层面解计算及绘图程序

### （一） P波初动断层面解计算程序

该程序是用FORTRAN-77语言编写的，根据P波初动计算断层面解。并自动在打印机或绘图仪上绘出图形。

（一）标识符说明

NS——粗选S值较大的个数(NS一般取15)。

MS——输入初动符号的台站数。

FL——震中经度$\lambda_e$(数据格式为：

118°40′25″写为118.040025)。

FA——震中纬度$\varphi_e$

DEPT——震源深度$h$(单位公里)。

AZI——AZ角的间隔(取30°)。

RI——R角的间隔(取15°)。

HI——H角的间隔(取15°)。

I,J,K——循环变量。

II——360°/AZI。

JJ——90°/RI。

KK——90°/HI。

CK——控制精度(一般取0.1°)。

ACK——节面可能转动范围的控制精度(一般取1.0°)。

AZ，R，H——震源坐标系（A，B，C）相对于固定坐标系（X，Y，Z)的夹角。

AZK,RK,HK——按角度分划时相应的AZ,R,H的角度值。

PIA——参数$P_i \cdot A$

PIC——参数$P_i \cdot C$

UP——震源球面上理论振幅分布，即PIA·PIC。

UPK——震源球面上第K个台的理论振幅分布。

AX，AY，AZZ——$A$的方向余弦

BX，BY，BZ——$B$的方向余弦。

CX，CY，CZ——$C$的方向余弦。

RD，DEG——度、分、秒与弧度互换系数。

AZD(7，2)——震源运动参数(方位角，倾向，倾角)的数组。

DA2(50)——输入初动符号的数组，符号规定如下：

$\bar{P}$写为+1或-1；$P^*$写为+2或-2；

$P_n$写为+3或-3；$P$写为+4或-4。其中+号表示初动向上，-号表示初动向下。

DA(50，3)——台站的信息 第一列为各台的经度$\lambda_s$，第二列为各台的纬度$\varphi_s$，第三列为台站的波型及初动标志。

DAT(50，3)——计算过程形成的台站有关参数，第一列为震中距$\Delta$，第二列为方位角Z，第三列为离源角$I_h$。

DA1(50，2)——输入各台站的经纬度，第一列为经度$\lambda_s$，第二列为纬度$\phi_s$。

S(1∶II，0∶JJ，0∶KK)——计算概率函数S(AZ，R，H)的数组。 AIH

SCR(4，179)——离源角表，第一行为震中距$\Delta$；第二行为震源在地壳上层相应的离源角；第三行为震源在地壳下层相应的离源角；第四行为震源在地幔中相应的离源角。

LINE(63，199)——字符型数组，供打印图形用。

(二) 子程序说明

RAD($E$)——度分秒化为弧度。

LAD($E$)——将地理纬度$E$化为地心纬度。

DIC(AZD，TD，HD)——根据AZD，RD，HD计算方向余弦AX，AY，AZZ，BX，BY，BZ，CX，CY，CZ。

FON(AZF，RF，HF，AZKF,RKF,HKF)——根据AZF，RF，HF及AZKF，RKF，HKF计算概率函数$S$。

UPI(AZU，RU，HU，ZU，RIH)——根据AZU，RU，HU，ZU，RIH等参数计算PIA，PIC，UP。

PAR——由AZ、R,H计算断层面及各力轴参数的无参过程。即计算$E_A$，$E_C$，DAA，DC，EP，$D_P$，$E_B$，$D_B$，$E_T$，$D_T$，MY，MC，MX，MA等震源运动参数。

PRDATA——计算各台站相应的震中距、方位角及离源角。

CALSCR——计算概率函数S，并粗选15个较大的S值。

SELSDR——对粗选的S值进行细选，并绘五个图。

（三）源程序的使用

1．数据分四段输入

（1）输入数组AIH；

（2）输入NS，MS，FL，FA，DEPT，AZI，RI，HI,CK，ACK；

（3）输入数组DA1；

（4）输入数组DA2。

2．计算及输出

（1）计算各台震中距、方位角及离源角并打印输出。

（2）按一定间隔分度计算 S 值，挑选出NS 个最大的 S 值并打印输出。

（3）对NS个最大的S值缩小间隔进行精选；从NS个精选值S中挑出最大的五个。

（4）由五组AZ，R，H分别计算震源运动参数。打印运动参数及相应的S，AZ，R，H值。

（5）将各个符号换算为相应的行列位置，置于二维字符型数组中，即可由打印机输出图形，包括初动符号、两个节面、五个力轴相应的坐标位置及坐标轴。

（四）FORTRAN-77语言计算程序

```
C
C     THE SOLUTION OF THE P-WAVE FIRST MOTION,
C     TO DETERMINE THE FOCAL MECHANISM OF EA-
C        RTHQUAKE
C
      INTEGER T, DA2
      COMMON/NM/NS, MS  T, II, JJ, KK
      COMMON/FD/FL,FA,DEPT, AZI, RI,HI, CK, ACK,
        AZ, R, H, AZK, RK, HK, PIA, PIC, UP, UPK,
        AX, AY, AZZ,
      BX, BY, BZ, CX, CY, CZ, RD, DEG
      COMMON/AD/AZD(7, 2), DA1(50, 2), DA2(50),
      DA(50, 3), DAT(50, 3), S(1:15, 0:10, 0:10), SCR
        (30, 4), AIH(4, 179)
C
C     INPUT THE INCIDENT ANGLE TABLE AND
C     THE DATA OF THE STATIONS
      OPEN(1, FILE='DATAIH')
      READ(1, *)((AIH(I, J), J=1, 179), I=1, 4)
      OPEN(2, FILE='MNF')
      READ(2, *)NS, MS, FL, FA, DEPT, AZI, RI,
      HI, CK, ACK
      II=IFI×(360./AZI+.01)
      JJ=IFI×(90./RI+.01)
      KK=IFI×(90./HI+.01)
      OPEN(3, FILE='DATA1')
      READ(3, *)((DA1(I, J), J=1, 2), I=1, MS)
      OPEN(4, FILE='DATA2')
      READ(4, *)(DA2(I), I=1, MS)
      DO 40 I=1, MS
      DA(I. 1)=DA1(I, 1)
```

```
      DA(I, 2)=DA1(I, 2)
      DA(I, 3)=DA2(I)
   40 CONTINUE
      WRITE(*, 50) MS, NS, FL, FA, DEPT, AZI, RI,
         HI, CK, ACK
   50 FORMAT(//, 2X, 'MS=', I4, 5X, 'NS=', I3, 5X,
      'FL=', F11.6,
      5X, 'FA=', F11.6, 5X, 'DEPTH=', F5.1, 5X, 'AZI
      =', F5.1,
      5X, 'RI=', F4.1, 5X, 'HI=', F4.1, 5X, 'CK=',
      F4.1, 5X, 'ACK=', F4.1, //)
      CALL PRDATA
      CALL CALSCR
      CALL SELSDR
      STOP
      END
C
C
      SUBROUTINE RAD(E)
      IF(E.LT.0)E=359.059060+E
      A=AINT(E)
      E=(E-A)*1000.
      B=AINT(E)
      E=(E-B)*1000.
      C=AINT(E)
      E=3.141592653*(A+B/60.+C/3600.)/180.
      RETURN
      END
C
C
      SUBROUTINE LAD(E)
```

```
      CALL RAD(E)
      E=(1.-1/298.25)**2*TAN(E)
      E=ATAN(E)
      E=3.141592653/2-E
      RETURN
      END
C
C
      SUBROUTINE DIC(AZD, TD, HD)
      COMMON/FD/FL, FA, DEPT, AZI, RI, HI,CK, ACK,
       AZ, R, H, AZK, RK, HK, PIA,
       PIC, UP, UPK, AX, AY, AZZ, BX, BY, BZ, CX,
       CY,CZ, RD, DEG
      AZO=AZD*RD
      RO=RD*TD
      HO=HD*RD
      CAZ=COS(AZO)
      CR=COS(RO)
      CH=COS(HO)
      SAZ=SIN(AZO)
      SR=SIN(RO)
      SH=SIN(HO)
      AX=SAZ*SR-CAZ*CR*SH
      AY=-(SH*SAZ*CR+CAZ*SR)
      AZZ=CH*CR
      BX=CH*CAZ
      BY=CH*SAZ
      BZ=SH
      CX=-(SH*CAZ*SR+SAZ*CR)
      CY=CAZ*CR-SH*SAZ*SR
      CZ=SR*CH
      RETURN
```

```
      END
C
C
      SUBROUTINE UPI(AZU, RU, HU, ZU, RIH)
      COMMON/FD/FL, FA, DEPT, AZI, RI, HI, CK,
         ACK, AZ, R, H, AZK, RK, HK, PIA,
         PIC, UP, UPK, AX, AY, AZZ, BX, BY, BZ,
         CX, CY, CZ, RD, DEG
      CALL DIC(AZU, RU, HU)
      ZO=ZU*RD
      RIHO=RIH*RD
      PIX=-SIN(RIHO)*COS(ZO)
      PIY=SIN(RIHO)*SIN(ZO)
      PIZ=-COS(RIHO)
      PIA=PIX*AX+PIY*AY+PIZ*AZZ
      PIC=PIX*CX+PIY*CY+PIZ*CZ
      UP=PIA*PIC
      RETURN
      END
C
C
      FUNCTION FON(AZF, RF, HF, AZKF, RKF, HKF
      INTEGER T, DA2
      COMMON/NM/NS, MS, T, II, JJ, KK
      COMMON/FD/FL, FA, DEPT, AZI, RI, HI, CK,
         ACK, AZ, R, H, AZK, RK, HK, PIA,
         PIC, UP, UPK, AX, AY, AZZ, BX, BY, BZ,
         CX, CY, CZ, RD, DEG
      COMMON/AD/AZD(7, 2), DA1(50, 2),DA2(50), DA
         (50, 3), DAT(50, 3),
         S(1:15, 0:10, 0:10), SCR(30, 4), AIH(4, 179)
      DIMENSION AM(4, 2)
```

```
      A=0.
      B=0.
      D=0.
      AK=0.
      DO 10 I=1, 4
      DO 10 J=1, 2
      AM(I, J)=0
10    CONTINUE
      DO 100 I=1, MS
      A2=DAT(I, 2)
      A3=DAT(I, 3)
      CALL UPI(AZKF, RKF, HKF, A2, A3)
      UPK=UP
      CALL UPI(AZF, RF, HF, A2, A3)
      IF((UP.EQ.0.).AND.(UPK.EQ.0.)) GO TO 80
      IF((UP.NE.0.).AND.(UPK.NE.0.)) GO TO 17
      GO TO 22
17    IF((UP/ABS(UP)).EQ.(UPK/ABS(UPK))) GO TO 80
22    IF((PIC.GE.0.).AND.(PIA.GE.0.)) GO TO 15
      GO TO 20
15    IF(DA(I, 3).GT.0) THEN
      AM(1, 1)=AM(1, 1)+1.
      ELSE
      AM(1, 2)=AM(1, 2)+1.
      END IF
20    CONTINUE
      IF((PIC.LT.0.).AND.(PIA.GE.0)) GO TO 30
      GO TO 40
30    IF(DA(I, 3).GT.0) THEN
      AM(2, 1)=AM(2, 1)+1.
      ELSE
      AM(2, 2)=AM(2, 2)+1.
```

```
      END IF
 40   CONTINUE
      IF((PIC.LT.0).AND.(PIA.LT.0.)) GO TO 50
      GO TO 60
 50   IF(DA(I, 3).GT.0.) THEN
      AM(3, 1)=AM(3, 1)+1.
      ELSE
      AM(3, 2)=AM(3, 2)+1.
      END IF
 60   IF(DA(I, 3).GT.0) THEN
      AM(4, 1)=AM(4, 1)+1.
      ELSE
      AM(4, 2)=AM(4, 2)+1.
      END IF
 80   CONTINUE
      WPI=2*UP*(1-ABS(UP))+0.3*(UP/ABS(UP))
      WPI=ABS(WPI)
      A=A+WPI*(UP/ABS(UP))*(DA(I, 3)/ABS(DA
         (I, 3)))
      B=B+WPI
      WPK=2*UPK*(1-ABS(UPK))+0.3*(UPK/ABS
         (UPK))
      WPK=ABS(WPK)
      AK=AK+WPK*(UPK/ABS(UPK))*(DA(I, 3)/ABS
      (DA(I, 3)))
100   CONTINUE
      DO 200 I=1, 4
200   D=D+0.3*AMIN1(AM(I, 1), AM(I, 2))
      IF(AK.LT.0) D=-D
      A=A+D
      A=ABS(A)
      IF(A.GT.B) A=B
```

```
      D=A/B
      FON=D
      RETURN
      END
C
C
      SUBROUTINE PAR
      REAL MX, MY, MZ, MA, MC, LX, LY
      INTEGER T, DA2
      COMMON/NM/NS, MS, T, II, JJ, KK
      COMMON/FD/FL, FA, DEPT, AZI, RI, HI, CK,
         ACK, AZ, R, H, AZK, RK, HK, PIA,
         PIC, UP, UPK, AX, AY, AZZ, BX, BY, BZ, CX,
         CY, CZ, RD, DEG
      COMMON/AD/AZD(7, 2), DA1(50, 2), DA2(50), DA
        (50, 3), DAT(50, 3),
      S(1:15, 0:10, 0:10), SCR(30, 4), AIH(4, 179)
      IF(AZZ.LT.0.) THEN
      AX=-AX
      AY=-AY
      AZZ=-AZZ
      END IF
      IF(CZ.LT.0.) THEN
      CX=-CX
      CY=-CY
      CZ=-CZ
      END IF
      IF(BZ.LT.0.) THEN
      BX=-BX
      BY=-BY
      BZ=-BZ
      END IF
```

```
IF(ABS(CY).LT.1E-10) THEN
EA=90.
ELSE
EA=ATAN(CX/CY)*DEG
END IF
IF(ABS(CZ).LT.1E-10) THEN
DAA=90
ELSE
DAA=ABS(1.-CZ*CZ)**0.5/CZ
DAA=ATAN(DAA)*DEG
END IF
IF(CY.LT.0.) THEN
EA=EA+270.
ELSE
EA=EA+90.
END IF
IF(ABS(AY).LT.1E-10) THEN
EC=90.
ELSE
EC=ATAN(AX/AY)*DEG
END IF
IF(ABS(AZZ).LT.1E-10) THEN
DC=90.
ELSE
DC=ABS(1-AZZ*AZZ)**0.5/AZZ
DC=ATAN(DC)*DFG
END IF
IF(AY.LT.0.) THEN
EC=EC+270.
ELSE
EC=EC+90.
END IF
```

```
IF(ABS(BY).LT.1E-10) THEN
EB=90
ELSE
EB=ATAN(BX/BR) * DEG
END IF
IF(ABS(BZ).LT.1E-10) THEN
DB=0
ELSE
DB=ABS(1.-BZ * BZ) * * 0.5/BZ
DB=ATAN(DB) * DEG
DB=90. -DB
END IF
IF(BY.LT.0. ) THEN
EB=EB+90.
ELSE
EB=EB+270.
END IF
IF(EB.LT.180.) THEN
EB=EB+180.
ELSE
EB=EB-180.
END IF
MX=0.70711 * (AX-CX)
MY=0.70711 * (AY-CY)
MZ=0.70711 * (AZZ-CZ)
PX=0.70711 * (AX+CX)
PY=0.70711 * (AY+CY)
PZ=0.70711 * (AZZ+CZ)
MA=MX * AX+MY * AY+MZ+AZZ
MC=MX * CX+MY * CY+MZ * CZ
WK=MA+MC
IF(MZ.LT.0.) THEN
```

```
MX = -MX
MY = -MY
MZ = -MZ
END IF
IF(PZ.LT.0.) THEN
PX = -PX
PY = -PY
PZ = -PZ
END IF
IF(ABS(MY).LT.1E-10) THEN
AZX = 90.
ELSE
AZX = ATAN(MX/MY) * DEG
END IF
IF(ABS(MZ).LT.1E-10) THEN
LX = 0.
ELSE
LX = ABS(1.-MZ * MZ) * * 0.5/MZ
LX = ATAN(LX) * DEG
LX = 90.-LX
END IF
IF(MY.LT.0.) THEN
AZX = AZX + 90.
ELSE
AZX = AZX + 270.
END IF
IF(ABS(PY).LT.1E-10) THEN
AZY = 90.
ELSE
AZY = ATAN(PX/PY) * DEG
END IF
IF(ABS(PZ).LT.1E-10) THEN
```

```
LY=0.
ELSE
LY=ABS(1.-PZ*PZ)**0.5/PZ
LY=ATAN(LY)*DEG
LY=90.-LY
END IF
IF(PY.LT.0.) THEN
AZY=AZY+90.
ELSE
AZY=AZY+270.
END IF
IF(WK.LT.0.) THEN
EP=AZX
DP=LX
ET=AZY
DT=LY
ELSE
EP=AZY
DP=LY
ET=AZX
DT=LX
END IF
IF(EP.LT.180.) THEN
EP=EP+180.
ELSE
EP=EP-180.
END IF
IF(ET.LT.180.) THEN
ET=ET+180.
ELSE
ET=ET-180.
END IF
```

```
      MY=EA
      MC=90.-DAA
      MX=EC
      MA=90.-DC
      IF(T.EQ.0) THEN
      WRITE(*, 10)
10    FORMAT(//, 5X, 'PLANE-A', 14X, 'PLANE-C',
         13X, 'AXIS-P', 11X,'
      AXIS-B', 12X, 'AXIS-T', 14X, 'AXIS-Y', 13X,
         'AXIS-X', //, 7X)
      WRITE(*, 15)
15    FORMAT(2X, 'AZIMUTH', 3X, 'DIP' 6X, 'AZIM-
         UTH', 3X, 'DIP', 7X, '
       AZIMUTH', 3X, 'DIP', 6X, 'AZIMUTH', 3X, 'DIP',
         7X, 'AZIMUTH',
      3X, 'DIP', 6X, A'ZIMUTH, 3X, 'DIP', 5X, 'AZIM-
         UTH', 3X, 'DIP', /)
       AZD(1, 1)=EA
       AZD(1, 2)=DAA
       AZD(2, 1)=EC
       AZD(2, 2)=DC
       AZD(3, 1)=EP
       AZD(3, 2)=DP
       AZD(4, 1)=EB
       AZD(4, 2)=DB
       AZD(5, 1)=ET
       AZD(5, 2)=DT
       AZD(6, 1)=MY
       AZD(6, 2)=MC
       AZD(7, 1)=MX
       AZD(7, 2)=MA
       END IF
```

```
      WRITE(*, 40)EA, DAA, EC, DC, EP, DP, EB,
         DB, ET, DT, MY, MC, MX, MA
40    FORMAT(1X, F7.2, 2X,F6.2, 5X,F6.2,3X,F5.2,7X,F
         6.2,2X,F5.2,6X,F6.2,
         3X,F5.2,5X,F6.2,3X,F5.2,5X,F6.2,3X,F5.2, 5X,
         F6.2, 3X, F5.2)
      RETURN
      END
C
      SUBROUTINE PRDATA
      INTEGER T, DA2,RR
      COMMON/NM/NS,MS,T,II,JJ,KK
      COMMON/FD/FL,FA,DEPT,AZI,RI,HI,CK,ACK,AZ,
         R,H,AZK,RK,HK,PIA,
      PIC,UP,UPK,AX,AY,AZZ,BX,BY,BZ,CX,CY,CZ,RD,
         DEG
      COMMON/AD/AZD(7,2),DA1(50,2),DA2(50),DA(50,3)
         DAT(50,3),
      S(1:15,0:10,0:10),SCR(30,4),AIH(4,179)
      WRITE(*,10)
10    FORMAT(8X,'STATION' ,14X,'LONGITUDE' ,11X,
        'LATITUDE' ,10X,'
        DELTA',13X,'AZIMUTH',12X,'IH' ,14X,'SIGN',//)
      CALL RAD(FL)
      CALL LAD(FA)
      RD=3.14159265/180.
      DEG=1./RD
      DO 200 J=1,MS
      SL=DA(J,1)
      SA=DA(J,2)
      CALL RAD(SL)
      CALL LAD(SA)
```

```
   CD=COS(FA)*COS(SA)+SIN(FA)*SIN(SA)*
      COS(FL-SL)
   SD=SQRT(1.-CD*CD)
   DD=ASIN(SD)
   DAT(J,1)=DD*DEG
   IF(CD.LT.0.)DAT(J,1)=180.-DAT(J,1)
   CZ=(COS(SA)-COS(FA)*CD)/SD/SIN(FA)
   SZ=SIN(SA)*SIN(SL-FL)/SD
   DAT(J,2)=ASIN(SZ)*DEG
   DAT(J,2)=ABS(DAT(J,2))
   IF((SZ.LE.0.).AND.(CZ.LE.0.))DAT(J,2)=180.
      +DAT(J,2)
     IF((SZ.LE.0.).AND.(CZ.GT.0.))DAT(J,2)=360.
      -DAT(J,2)
   IF((SZ.GT.0).AND.(CZ.LE0))DAT(J,2)=180DAT(J,2)
   IF(ABS(DA(J,3)).EQ.1.)DAT(J,3)=ATAN
      (DD*6371.221/DEPT)*DEG
   IF(DAT(J,1).LT.2.) THEN
   IF(ABS(DA(J,3)).EQ.4.) THEN
   DAT(J,3)=ATAN(DD*6371.221/DEPT)*DEG
   END IF
   IF(ABS(DA(J,3)).EQ.2.)DAT(J,3)=ASIN(6./6.75)
      *DEG
   IF(ABS(DA(J,3)).EQ.3.) GO TO 20
   GOTO 30
20 IF(DEPT.LT.15.) THEN
   DAT(J,3)=ASIN(6.0/8.049)*DEG
   ELSE
   DAT(J,3)=ASIN(6.75/8.049)*DEG
   END IF
30 CONTINUE
   ELSE IF(DAT(J,1).LE.100.) THEN
```

```
60  M=M+1
    IF(AIH(1,M).LT.DAT(J,1)) GO TO 60
    DAT(J,3)=0.
    IF((M-1).LE.1) M=2
    IF((M-1).GE.176) M=177
    IF(DEPT.LT.15.) THEN
    RR=2
    ELSE IF(DEPT.LT.40.) THEN
    RR=3
    ELSE
    RR=4
    END IF
    DO 90 I=M-1,M+2
    C=AIH(RR,I)
    DO 70 K=M-1,I-1
70  C=C*(DAT(J,1)-AIH(1,K))/(AIH(1,I)-AIH(1,K))
    DO 80 K=I+1,M+2
80  C=C*(DAT(J,1)-AIH(1,K))/(AIH(1,I)-AIH(1,K))
    DAT(J,3)=DAT(J,3)+C
90  CONTINUE
    ELSE
    DAT(J,3)=16.0
    END IF
    A1=DA(J,1)
    A2=DA(J,2)
    A3=DAT(J,1)
    A4=DAT(J,2)
    A5=DAT(J,3)
    A6=DA(J,3)
    IF(A6.EQ.1.) THEN
    WRITE(*,130)J,A1,A2,A3,A4,A5
```

```
130  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2,F19.2,F19.2,
       15X,3H+PD)
     ELSE IF(A6.EQ.-1) THEN
     WRITE(*, 131)J, A1, A2, A3, A4, A5
131  FORMAT(10X,I2,2X,F24.6,F20.6, F14.2,F19.2,F19.2'
       15X,3H-PD)
     ELSE IF(A6.EQ.2.) THEN
     WRITE(*,132)J,A1,A2,A3,A4,A5
132  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2,F19.2,F19.2,
       15X,3H+P*)
     ELSE IF(A6.EQ.-2.) THEN
     WRITE(*,133)J,A1,A2,A3,A4,A5
133  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2, F19.2,F19.2'
       15X,3H-P*)
     ELSE IF(A6.EQ.3.) THEN
     WRITE(*,134)J,A1,A2,A3,A4,A5
134  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2,F19.2,F19.2,
       15X,3H+PN)
     ELSE IF(A6.EQ.-3.) THEN
     WRITE(*,135)J,A1,A2,A3,A4,A5
135  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2,F19.2,F19.2 ,
       15X,3H-PN)
     ELSE IF(A6.EQ,4.) THEN
     WRITE(*,136)J,A1,A2,A3A4,A5,
136  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2,F19.2,F19.2,
       15X,2H+P) ELSE
     WRITE(*,137)J,A1,A2,A3,A4,A5
137  FORMAT(10X,I2,2X,F24.6,F20.6,F14.2,F19.2,F19.2,
       15, X,2H-P)
     END IF
200  CONTINUE
     RETURN
```

```
      END
C
C
      SUBROUTINE CALSCR
      INTEGER T,DA2,B,C,E
      COMMON/NM/NS,MS,T,II,JJ,KK
      COMMON/FD/FL,FA,DEPT,AZI,RI,HI,CK,ACK,
        AZ,R,H,AZK,RK,HK,
      PIA,PIC,UP,UPK,AX,AY,AZZ,BX,BY,BZ,CX,CY,
        CZ,RD,DEG
      COMMON/AD/AZD(7,2),DA1(50,2),DA2(50),DA
        (50,3),DAT(50,3),
      S(1:15,0:10,0:10),SCR(30,4),AIH(4,179)
      DO 10 I=1,II
      DO 10 J=0,JJ
      DO 10 K=0,KK
      AZ=AZI * FLOAT(I)
      R=RI * FLOAT(J)
      H=HI * FLOAT(K)
      IF(K.LT.KK) THEN
      AZK=AZ
      RK=R
      HK=HI * FLOAT(K+1)
      ELSE IF(J.LT.JJ) THEN
      AZK=AZ
      RK=RI * FLOAT(J+1)
      HK=H
      ELSE IF(I.LT.II) THEN
      AZK=AZI * FLOAT(I+1)
      RK=R
      HK=H
      ELSE
```

```
      AZK = AZI
      RK = R
      HK = H
      END IF
      S(I,J,K) = FON(AZ,R,H,AZK,RK,HK)
   10 CONTINUE
      DO 50 L = 1,NS
      A = 0.
      DO 20 I = 1,II
      DO 20 J = 0,JJ
      DO 20 K = 0,KK
      IF(S(I,J,K).GT.A) THEN
      A = S(I,J,K)
      B = I
      C = J
      E = K
      END IF
   20 CONTINUE
      SCR(L,1) = AZI * FLOAT(B)
      SCR(L,2) = RI * FLOAT(C)
      SCR(L,3) = HI * FLOAT(E)
      SCR(L,4) = A
      S(B,C,E) = 0.
   50 CONTINUE
      WRITE( * ,100)((SCR(I,J),J = 1,4),I = 1,NS)
  100 FORMAT(5X, 'SCR' ,/,(10X,8F12.4))
      RETURN
      END
C
C
      SUBROUTINE SELSDR
      INTEGER T,DA2,P,Q
      COMMON/NM/NS,MS,T,II,JJ,KK
```

```
      COMMON/FD/FL, FA,DEPT,AZI,RI,HI,CK,ACK,A
         Z,R,H,AZK,RK,HK,
      PIA,PIC,UP,UPK,AX,AY,AZZ,BX,BY,BZ,CX,CY,
         CZ,RD,DEG
      COMMON/AD/AZD(7,2),DA1(50,2),DA2(50),DA(50,
         3),DAT(50,3),
         S(1:15,0:10,0:10),SCR(30,4),AIH(4,179)
      DIMENSION AG(8,3),DC(3,3),DCC(3,3),FC(6)
      CHARACTER LINE(63,179)
      SMA = 0.
      DO 100 I = 1,NS
      AZJ = AZI
      RJ = RI
      HJ = HI
10    K = 0
      A = SCR(I,4)
      AZK = SCR(I,1)
      RK = SCR(I,2)
      HK = SCR(I,3)
20    DO 30 J = 1,4
      AG(J,1) = AZK + AZJ
      AG(J + 4,1) = AZK - AZJ
      AG(2 * J - 1,3) = HK + HJ
      AG(2 * J,3) = HK - HJ
30    CONTINUE
      DO 40 J = 1,2
      AG(J + 4,2) = RK + RJ
      AG(J,2) = AG(J + 4,2)
      AG(J + 6,2) = RK - RJ
      AG(J + 2,2) = AG(J + 6,2)
40    CONTINUE
      DO 50 J = 1,8
```

```
      AZ=AG(J,1)
      R=AG(J,2)
      H=AG(J,3)
      IF(H.GT.90.) THEN
      H=180.-H
      AZ=180.+AZ
      END IF
      IF(H.LT.0) THEN
      H=-H
      AZ=180.+AZ
      END IF
      AG(J,1)=AZ
      AG(J,2)=R
      AG(J,3)=H
      W=FON(AZ,R,H,AZK,RK,HK)
      IF(W.GT.A) THEN
      A=W
      K=J
      END IF
50    CONTINUE
      IF(K.NE.0) THEN
      SCR(I,1)=AG(K,1)
      SCR(I,2)=AG(K,2)
      SCR(I,3 =AG(K,3)
      SCR(I,4)=A
      GO TO 10
      ELSE IF(AZJ.GT.CK) THEN
      AZJ=AZJ/2.
      RJ=RJ/2.
      HJ=HJ/2.
      GO TO 20
```

```
      END IF
      B=SCR(I,1)
      C=SCR(I,2)
      D=SCR(I,3)
100   CONTINUE
      K=0
      DO 1000 II=1,5
110   K=K+1
      SMA=0.
      DO 120 I=1,NS
      IF(SCR(I,4).GT.SMA) THEN
      SMA=SCR(I,4)
      J=I
      END IF
120   CONTINUE
      IF((II.GT.1).AND.(IFI×(SCR(J,3)*100).EQ.IFI
         ×(HM*100.)))
      GO TO 180
150   IF((IFI×(SCR(J,1)*100.).EQ.IFI×(AZM*100.))
      AND.(IFI×(SCR(J,2)*100.).EQ.IFI×(RM*100.)))
         GO TO 160
      GO TO 180
160   SCR(J,4)=0.
      IF(K.EQ.N) THEN
      GO TO 1000
      ELSE
      GO TO 110
      END IF
180   CONTINUE
      AZM=SCR(J,1)
      RM=SCR(J,2)
      HM=SCR(J,3)
```

```
      SCR(J,4)=0.
      WRITE(*,2000)SMA,AZM,RM,HM
2000  FORMAT(/,10X,'SMAX=',F9.6,5X,'AZMAX=',
         F7.2,5X,'RMAX=',F7.2,5X,'HMAX=',
         F7.2,//)
      DA(II,1)=AZM
      DA(II,2)=RM
      DAT(II,1)=HM
      CALL DIC(AZM,RM,HM)
      DC(1,1)=AX
      DC(1,2)=AY
      DC(1,3)=AZZ
      DC(2,1)=BX
      DC(2,2)=BY
      DC(2,3)=BZ
      DC(3,1)=CX
      DC(3,2)=CY
      DC(3,3)=CZ
      T=0
      CALL PAR
      DC 300 I=1,6
      ACK=-ACK
      AK=ACK
      AF=AK
      T=1
220   J=IFI×((I+1.1)/2.)
      IF(J.EQ.1) THEN
      K=2
      L=3
      END IF
      IF(J.EQ.2) THEN
      K=1
```

```
      L=3
      END IF
      IF(J.EQ.3) THEN
      K=1
      L=2
      END IF
      DO 230 M1=1,3
      DCC(J,M1)=DC(J,M1)
      DCC(L,M1)=DC(K,M1) * SIN(AF)+DC(L,M1)*COS
         (AF)
      DCC(K,M1)=DC(K,M1)*COS(AF)-DC(L,M1) *SIN
         (AF)
230   CONTINUE
      Q=0
      DO 240 P=1,MS
      AZ=DAT(P,2)*RD
      H=DAT(P,3)*RD
      PIX=-SIN(H) * COS(AZ)
      PIY=SIN(H) * SIN(AZ)
      PIZ=-COS(H)
      PIA=PIX * DC(1,1)+PIY * DC(1,2)+PIZ * DC(1,3)
      PIC=PIX * DC(3, 1) +PIY * DC(3,2)+PIZ * DC(3,3
      UP=PIA * PIC
      PIA=PIX * DCC(1,1)+PIY * DCC(1,2)+PIZ * DCC
         (1,3)
      PIC=PIX * DCC(3,1)+PIY * DCC(3,2) +PIZ * DCC
         (3,3)
      UPK=PIA * PIC
      IF((UP.EQ.0.).AND.(UPK.EQ.0.)) GO TO 240
      IF((UP.NE.0.).AND.(UPK.NE.0.)) GO TO 19
      Q=Q+1
      GO TO 240
```

```
 19  IF((UP/ABS(UP)).NE.(UPK/ABS(UPK)))Q=Q+1
240  CONTINUE
     IF(Q.GT.2) GOTO 260
     IF(T.EQ.1) THEN
     AF=AF+AK
     ELSE
     AF=AF+AK/2.
     AK=AK/2.
     END IF
     IF(ABS(AK).GT.0.02) GO TO 220
     GO TO 280
260  AF=AF-AK/2.
     AK=AK/2.
     T=2
     IF(ABS(AK).GT.0.02) GO TO 220
280  AX=DCC(1,1)
     AY=DCC(1,2)
     AZZ=DCC(1,3)
     BX=DCC(2,1)
     BY=DCC(2,2)
     BZ=DCC(2,3)
     CX=DCC(3,1)
     CY=DCC(3,2)
     CZ=DCC(3,3)
     CALL PAR
     FC(I)=AF
300  CONTINUE
     WRITE(*,310)
310  FORMAT(//,10X,'ANGLE OF ROTATION',//)
     WRITE(*,320)(FC(I),I=1,6)
320  FORMAT(15X,6F10.2,/)
     DO 330 I=1,63
```

```
      DO 330 J=1,119
 330  LINE(I,J)='
      DO 331 J=1,119
 331  LINE(32,J)='-'
      DO 332 I=1,63
 332  LINE(I,60)='|'
      LINE(32,60)='O'
      LINE(1,60)='N'
      DO 350 JJ=1,MS
      AZ=DAT(JJ,2)*RD
      H=DAT(JJ,3)*RD
      AX=SIN(AZ)*SIN(H)/(1.+COS(H))
      AY=COS(AZ)*SIN(H)/(1.+COS(H))
      I=32-IFI×(30.*AY+0.5)
      J=60+IFI×(58.5*AX+0.5)
      IF(DA(JJ,3).GT.0.) THEN
      LINE(I,J)='+'
      ELSE
      LINE(I,J)='/'
      END IF
 350  CONTINUE
      IF(AZD(1,1).LT.180.) THEN
      AZD(1,1)=AZD(1,1)+180.
      ELSE
      AZD(1,1)=AZD(1,1)-180.
      END IF
      IF(AZD(2,1).LT.180.) THEN
      AZD(2,1)=AZD(2,1)+180.
      ELSE
      AZD(2,1)=AZD(2,1)-180.
      END IF
      DO 400 JJ=1,7
```

```
AZ=AZD(JJ,1)*RD
H=AZD(JJ,2)*RD
AX=SIN(AZ)*COS(H)/(1.+SIN(H))
AY=COS(AZ)*COS(H)/(1.+SIN(H))
I=32-IFI×(30.*AY+0.5)
J=60+IFI×(58.5*AX+0.5)
LINE(I,J)='O'
IF(JJ.EQ.1) THEN
LINE(I,J+1)='A'
ELSE IF(JJ.EQ.2) THEN
LINE(I,J-1)='C'
ELSE IF(JJ.EQ.3) THEN
LINE(I+1,J)='P'
ELSE IF(JJ.EQ.4) THEN
LINE(I+1,J)='B'
ELSE IF(JJ.EQ.5) THEN
LINE(I+1,J)='T'
ELSE IF(JJ.EQ.6) THEN
LINE(I,J)='Y'
ELSE IF(JJ.EQ.7) THEN
LINE(I,J)='X'
END IF
IF((JJ.EQ.1).OR.(JJ.EQ.2)) THEN
CZ=AZ-90.*RD
BX=SIN(CZ)
BY=COS(CZ)
I=32-IFI×(30.*BY+0.5)
J=60+IFI×(58.5*BX+0.5)
LINE(I,J)='O'
I=64-I
J=120-J
LINE(I,J)='O'
```

```
      END IF
  400 CONTINUE
      WRITE(*,500) ((LINE(I,J),J=1,119),I=1,63)
  500 FORMAT(10X,119A1)
 1000 CONTINUE
      RETURN
      END
```

## (二) P波初动断层面解绘图程序

本程序根据前述计算结果，利用绘图仪绘出五个不同的P 波初动解节面分布图。

**(一) 标识符说明**

AZD(7, 2)——震源运动参数数组

DAT(50, 2)——台站有关参数，第一列为方位角，第二列为离源角。

X(6, 2)——关于节面 A的信息。

Y(6, 2)——关于节面 C的信息。

IDA(50)——*P* 波初动符号组。

LINE, NC, MCH, LC——控制变量。

MS——有关初动符号的台站数。

**(二) 子程序说明**

PLOTS(J, IS)——开启或关闭绘图仪的子程序。

CHENGP(J, IS)——换不同色笔的子程序。

PLOT(XO, YO, I, IS)——将绘图笔移至某点的子程序。

SYMBOL(X, Y, IANG, K, NCH, IS)——绘字符子程序。

ARC(XCOR, YCOR, RO, R1, IA1, IA2, IS)——画弧子程序。

XJZ(X, XA, K)——形成增广矩阵，化为正规方程组，求

出绘节面各相应参数。

QATA(N，A，AT)——化增广矩阵为正规方程组。

QG(N,XX,YY,A)——高斯主元消去法解方程

HW(QX,IQD,IZD,XX,YY,R)——求出二节面相应的起始及终结的角度值。

**（三）源程序使用说明**

数据输入。通过数据文件(一般在计算过程中形成)，分别输入初动符号组 IDA，台站参数DAT，及震源参数AZD。

为了进行比较，可输入五组不同的震源运动参数，将它们分别放在五个文件中：COMM6，COMM7，COMM8，COMM9，COMM10中，依次执行这些文件内容，即可绘出五张不同的图形。

**（四）源程序**

```
      DIMENSION AZD(7,2),DAT(50,2)
      DIMENSION X(6,2),Y(6,2),IDA(50)
      CHARACTER LINE,NC,MCH,LC
      PAUSE'INPUT MS'
      READ*,MS
      RD=3.141592653/180.
      OPEN(6,FILE='PT')
      READ(6,*)(IDA(J),J=1,MS)
      OPEN(12,FILE='DEAR')
      READ(12,*) ((DAT(I,J),J=1,2),I=1,MS)
      OPEN(1,FILE='ANY1')
      OPEN(2,FILE='ANY2')
      OPEN(3,FILE='ANY3')
      OPEN(4,FILE='ANY4')
      OPEN(5,FILE='ANY5')
      DO 1000 II=1,5
      READ(II,*) ((AZD(I,J),J=1,2),I=1,7)
```

```
IF(II.EQ.1) THEN
MCH='1'
ELSE IF(II.EQ.2) THEN
MCH='2'
ELSE IF(II.EQ.3) THEN
MCH='3'
ELSE IF(II.EQ.4) THEN
MCH='4'
ELSE
MCH='5'
END IF
NC='N'
LC='0'
OPEN (7,FILE='COMM6' ,STATUS='NEW')
OPEN (8,FILE='COMM7' ,STATUS='NEW')
OPEN (9,FILE='COMM8' ,STATUS='NEW')
OPEN (10,FILE='COMM9' ,STATUS='NEW')
OPEN (11,FILE='COMM10' ,STATUS='NEW')
CALL PLOTS(1,II+6)
CALL PLOT(18.0,14.0,-3,II+6)
CALL ARC(0.0,0.0,10.0,10.0,0,360,II+6)
CALL SYMBOL(0.,0.,0,5,LC,II+6)
CALL CHENGP(2,II+6)
DO 350 JJ=1,26
AZ=DAT(JJ,1)*RD
H=DAT(JJ,2)*RD
AX=SIN(AZ)*SIN(H)*10./(1.+COS(H))
AY=COS(AZ)*SIN(H)*10./(1.+COS(H))
IF(IDA(JJ).GT.0) THEN
LINE='+'
ELSE
LINE='-'
```

```
      END IF
      CALL SYMBOL (AX,AY,0,5,LINE,II+6)
350   CONTINUE
      CALL CHENGP(3,II+6)
      XA=AZD(1,1)
      IF(AZD(1,1).LT.180.) THEN
      AZD(1,1)=AZD(1,1)+180.
      ELSE
      AZD(1,1)=AZD(1,1)-180.
      END IF
      XC=AZD(2,1)
      IF(AZD)(2,1).LT.180.) THEN
      AZD(2,1)=AZD(2,1)+180.
      ELSE
      AZD(2,1)=AZD(2,1)-180.
      END IF
      DO 400 JJ=1,7
      AZ=AZD(JJ,1)*RD
      H=AZD(JJ,2)*RD
      AX=SIN(AZ)*COS(H)*10./(1.+SIN(H))
      AY=COS(AZ)*COS(H)*10./(1.+SIN(H))
      IF(JJ.EQ.1) THEN
      X(JJ,1)=AX
      X(JJ,2)=AY
      LINE='A'
      ELSE IF(JJ.EQ.2) THEN
      Y(JJ-1,1)=AX
      Y(JJ-1,2)=AY
      LINE='C'
      ELSE IF(JJ.EQ.3) THEN
      LINE='P'
      ELSE IF(JJ.EQ.4) THEN
```

```
LINE = 'B'
X(JJ,1) = AX
X(JJ,2) = AY
Y(JJ,1) = AX
Y(JJ,2) = AY
ELSE IF(JJ.EQ.5) THEN
LINE = 'T'
ELSE IF(JJ.EQ.6) THEN
LINE = 'Y'
Y(JJ - 1,1) = AX
Y(JJ - 1,2) = AY
ELSE IF(JJ.EQ.7) THEN
LINE = 'X'
X(JJ - 2,1) = AX
X(JJ - 2,2) = AY
END IF
CALL SYMBOL(AX,AY,0,5,LINE,II + 6)
IF((JJ.EQ.1).OR.(JJ.EQ.2)) THEN
CZ = AZ - 90. * RD
BX = 10. * SIN(CZ)
BY = 10. * COS(CZ)
LINE = '0'
IF(JJ.EQ.1) THEN
X(JJ + 1,1) = BX
X(JJ + 1,2) = BY
X(JJ + 2,1) = - BX
X(JJ + 2,2) = - BY
ELSE IF(JJ.EQ.2) THEN
Y(JJ,1) = BX
Y(JJ,2) = BY
Y(JJ + 1,1) = - BX
Y(JJ + 1,2) = - BY
```

```
      END IF
      CALL SYMBOL(BX,BY,0,5,LINE,II+6)
      CALL SYMBOL(-BX,-BY,0,5,LINE,II+6)
      END IF
  400 CONTINUE
      CALL SYMBOL(0.,10.1,0,5,NC,II+6)
      CALL CHENGP(1,II+6)
      CALL SYMBOL(0.,12.,0,8,MCH,II+6)
      CALL CHENGP(3,II+6)
      CALL XJZ(X,XA,II+6)
      CALL XJZ(Y,XC,II+6)
      CALL PLOTS(2,II+6)
 1000 CONTINUE
      STOP
      END
C
      SUBROUTINE ARC(XCOR,YCOR,R0,R1,IA1,IA2,IS)
      IX=XCOR*100.
      IY=YCOR*100.
      IR=R0*100
      IR1=R1*100
      WRITE(IS,*)'AC; ',IX,',',IY,',',IR,',',IR1,',',
       IA1,',',IA2
      RETURN
      END
      SUBROUTINE PLOT(X0,Y0,I,IS)
      PRINT*,'X0,Y0'
      WRITE(*,*) X0,Y0
      IX=X0*100
      IY=Y0*100
      WRITE(IS,*) 'AP; ',IX,',',IY,',',I
      RETURN
      END
```

```
      SUBROUTINE PLOTS(J,IS)
      IF(J.EQ.2) GO TO 15
      WRITE(IS,*) 'EX; A'
      WRITE(IS,*) 'IN'
      GO TO 25
15    WRITE(IS,*) 'HM'
      WRITE(IS,*) 'TE'
25    RETURN
      END
      SUBROUTINE SYMBOL(X,Y,IANG,K,NCH,IS)
      CHARACTER NCH
      IX=X*100
      IY=Y*100
      WRITE(IS,*) 'AS; ',IX,',',IY,',',K,',',IANG,',',
         NCH
      RETURN
      END
      SUBROUTINE XJZ(X,XA,K)
      DIMENSION X(6,2),A(11,4),AT(3,4)
      F=(X(1,1))**2+(X(1,2))**2
      DO 11 I=1,4
      A(I,1)=X(I+1,1)-X(1,1)
      A(I,2)=X(I+1,2)-X(1,2)
      A(I,3)=(X(I+1,1))**2+(X(I+1,2))**2-F
11    A(I,3)=A(I,3)/2.
      CALL QATA(4,A,A,T)
      CALL QG(2,XX,YY,AT)
      PRINT*,'XX--YY'
      WRITE(*,*)XX,YY
      RR=0.
      DO 112 I=1,5
      P=X(X-X(I,1))**2+(YY-X(I,2))**2
```

```
      P=SQRT(P)
112   RR=RR+P
      RR=RR/5.
      PRINT*,'RR'
      WRITE(*,*)RR
      CALL PLOT(XX,YY,-3,K)
      CALL HW(XA,IQD,IZD,XX,YY,RR)
      CALL ARC(0.,0.,RR,RR,IQD,IZD,K)
      CALL PLOT(-XX,-YY,-3,K)
      RETURN
      END
      SUBROUTINE HW(QX,IQD,IZD,XX,YY,R)
      DEG=180./3.141592653
      ALPH=ASIN(10./R)*DEG
      PRINT* 'XX&YY'
      WRITE(*,*) XX,YY,QX
      BLTE=ATAN(ABS(YY/XX))*DEG
      PRINT*,'ALPH,BLTE'
      WRITE(*, *)ALPH, BLTE
      IF((QX.GE.0.).AND.(QX.LE.90.)) THEN
      IQD=180.-ALPH+BLTE+0.5
      IZD=180.+ALPH+BLTE+0.5
      ELSE IF((QX.GT.90.).AND.(QX.LE.180.)) THEN
      IQD=180.-ALPH-BLTE+0.5
      IZD=180.+ALPH-BLTE+0.5
      ELSE IF((QX.GT.180.).AND.(QX.LT.270.)) THEN
      IQD=360.-(ALPH-BLTE)+0.5
      IZD=360.+ALPH+BLTE+0.5
      ELSE IF((QX.GE.270.).AND.(QX.LT.360.)) THEN
      IQD=360.-ALPH-BLTE+0.5
      IZD=360.+ALPH-BLTE+0.5
      END IF
```

```
      PRINT*,'IQD,IZD'
      WRITE(*,*)IQD,IZD
      RETURN
      END
      SUBROUTINE QATA(N,A,AT)
      DIMENSION A(11,4),AT(3,4)
      DO 12 I=1,2
      DO 12 J=1,3
      AT(I,J)=0.
      DO 12 K=1,N
12    AT(I,J)=AT(I,J)+A(K,I)+A(K,J)
      PRINT*,'AT'
      WRITE(*,13) ((AT(I,J),J=1,3), I=1,2)
13    FORMAT(1X,3F10.4)
      RETURN
      END
      SUBROUTINE QG(N,XX,YY,A)
      DIMENSION A(3,4)
      NM1=N-1
      DO 10 K=1,NM1
      C=0
      DO 2 I=K,N
      IF(ABS(A(I,K)).LE.ABS(C)) GO TO 2
      C=A(I,K)
      I0=I
 2    CONTINUE
      IF(ABS(C).GE.1E-06) GO TO 3
      ISW=0
      RETURN
 3    IF(I0.EQ.K) GO TO 6
      DO 4 J=K,N
      S=A(K.J)
```

```
      A(K, J)=A(I0, J)
  4   A(I0,J)=S
      S =A(K,3)
      A(K,3)=A(I0,3)
      A(I0,3)=S
  6   KP1=K+1
      C=1/C
      A(K,3)=A(K,3)*C
      DO 10 J=KP1,N
      A(K,J)=A(K,J)*C
      DO 20 I=KP1,N
 20   A(I,J)=A(I,J)-A(I,K)*A(K,J)
 10   A(J,3)=A(J,3)-A(J,K)*A(K,3)
      A(N,3)=A(N,3)/A(N,N)
      DO 40 K=1,NM1
      I=N-K
      C=0
      IP1=I+1
      DO 50 J=IP1,N
 50   C=C+A(I,J)*A(J,3)
 40   A(I,3)=A(I,3)-C
      YY=A(2,3)
      XX=A(1,3)
      PRINT*, 'XX,YY'
      WRITE(*,*) XX,YY
      ISW=1
      RETURN
      END
      SUBROUTINE CHENGP(J,IS)
      WRITE(IS,*) 'NP; ' ,J
      RETURN
      END
```

# 附录 计算实例

某次地震有26个台站的资料，其计算结果及打印机输出图形如下(并附有离源角表)：

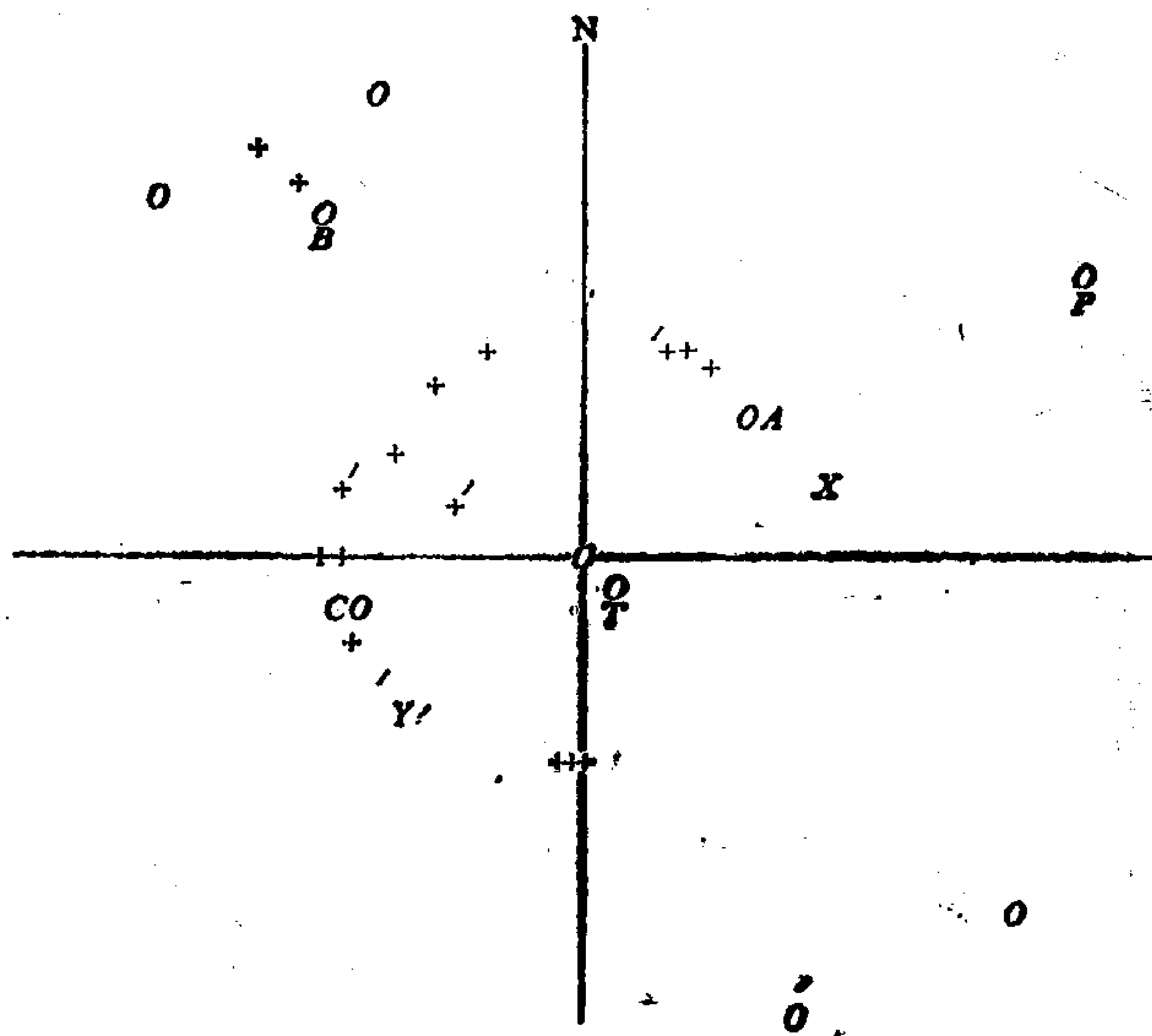



图4-附1

MS=26 NS=15 FL=119.015000 FA=31.027010 DEPTH=10.0 AZI=30.0 RI=15.0 HI=15.0 CK=.1 ACK=1.0

| STATION (台站编号) | LONGITUDE (经度) | LATITUDE (纬度) | DELTA (震中距) | AZIMUTH (方位角) | IH (离源角) | SIGN (初动符号) |
|---|---|---|---|---|---|---|
| 1 | 118.015100 | 32.048640 | 1.85 | 325.56 | 87.22 | +P |
| 2 | 116.016000 | 31.020050 | 2.77 | 268.75 | 47.81 | +P |
| 3 | 120.006000 | 30.010000 | 1.38 | 157.88 | 86.28 | −P |
| 4 | 87.041040 | 43.049020 | 27.88 | 305.31 | 28.87 | −P |
| 5 | 75.058050 | 39.027020 | 36.07 | 295.07 | 27.54 | +P |
| 6 | 119.016000 | 26.005030 | 5.35 | 182.24 | 47.67 | +P |
| 7 | 119.000000 | 25.026050 | 6.00 | 184.31 | 47.60 | +P |
| 8 | 118.010000 | 26.038660 | 4.77 | 194.60 | 47.72 | −P |
| 9 | 110.001100 | 40.034040 | 11.90 | 322.53 | 47.02 | +P |
| 10 | 118.047000 | 32.003050 | .73 | 326.91 | 82.94 | +P |
| 11 | 112.055000 | 28.010060 | 6.58 | 241.97 | 47.54 | +P |
| 12 | 109.019000 | 19.042000 | 14.66 | 221.12 | 45.79 | −P |
| 13 | 108.055000 | 34.002020 | 9.28 | 288.93 | 47.27 | +P |
| 14 | 108.028000 | 22.046010 | 13.08 | 231.30 | 46.66 | −P |
| 15 | 119.027000 | 31.022060 | .08 | 212.91 | 41.34 | −P |
| 16 | 103.050000 | 36.005020 | 13.82 | 293.75 | 46.30 | −P |
| 17 | 125.018000 | 43.049040 | 13.18 | 18.72 | 46.62 | +P |
| 18 | 118.035000 | 24.056030 | 6.54 | 187.36 | 47.55 | +P |
| 19 | 123.034000 | 41.049040 | 10.86 | 16.35 | 47.11 | −P |
| 20 | 99.048050 | 39.024040 | 17.89 | 301.77 | 41.70 | +P |
| 21 | 102.044000 | 25.007020 | 16.06 | 251.05 | 44.32 | −P |
| 22 | 124.018100 | 40.003030 | 9.43 | 23.11 | 47.26 | +P |
| 23 | 117.007000 | 36.012040 | 5.14 | 337.92 | 47.69 | +P |
| 24 | 101.058000 | 30.003030 | 15.15 | 269.28 | 45.36 | +P |
| 25 | 129.035600 | 44.036060 | 15.40 | 28.57 | 45.10 | +P |
| 26 | 121.011000 | 31.005050 | 1.48 | 103.47 | 86.52 | −P |

SCR

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 180.0000 | 60.0000 | 30.0000 | .5294 | 210.0000 | 45.0000 | 15.0000 | .5215 |
| 150.0000 | .0000 | 60.0000 | .5001 | 150.0000 | 90.0000 | 60.0000 | .5001 |
| 30.0000 | 45.0000 | .0000 | .4931 | 120.0000 | 90.0000 | 45.0000 | .4694 |
| 90.0000 | 30.0000 | 30.0000 | .4587 | 330.0000 | 90.0000 | 90.0000 | .4545 |
| 180.0000 | 60.0000 | 60.0000 | .4493 | 30.0000 | 60.0000 | 30.0000 | .4438 |
| 150.0000 | 90.0000 | 90.0000 | .4338 | 240.0000 | 90.0000 | 90.0000 | .4338 |
| 60.0000 | 90.0000 | 90.0000 | .4338 | 180.0000 | 60.0000 | 75.0000 | .4286 |
| 150.0000 | 90.0000 | 75.0000 | .4278 | | | | |

SMAX= .618093 AZMAX= 213.75 RMAX= 43.13 HMAX= 13.12

| PLANE-A | | PLANE-C | | AXIS-P | | AXIS-B | | AXIS-T | | AXIS-Y | | AXIS-X | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP |
| 224.24 | 48.26 | 69.88 | 44.70 | 56.68 | 1.83 | 326.25 | 13.12 | 154.45 | 76.75 | 224.24 | 41.74 | 69.88 | 45.30 |
| 221.98 | 48.83 | 69.88 | 44.70 | 55.45 | 2.13 | 324.90 | 14.35 | 153.70 | 75.49 | 221.98 | 41.17 | 69.88 | 45.30 |
| 241.05 | 45.64 | 69.88 | 44.70 | 65.43 | .47 | 335.39 | 4.43 | 161.50 | 85.55 | 241.05 | 44.36 | 69.88 | 45.30 |
| 224.95 | 49.97 | 70.73 | 43.01 | 57.08 | 3.57 | 326.25 | 13.12 | 162.00 | 76.38 | 224.95 | 40.03 | 70.73 | 46.99 |
| 221.86 | 43.17 | 67.61 | 49.81 | 235.46 | 3.41 | 326.25 | 13.12 | 131.19 | 76.43 | 221.86 | 46.83 | 67.61 | 40.19 |
| 224.24 | 48.26 | 30.66 | 42.55 | 37.79 | 2.88 | 128.13 | 6.81 | 285.01 | 82.61 | 224.24 | 41.74 | 30.66 | 47.45 |
| 224.24 | 48.26 | 115.11 | 69.83 | 264.33 | 13.12 | 6.20 | 41.43 | 160.56 | 45.61 | 224.24 | 41.74 | 115.11 | 20.17 |

ANGLE OF ROTATION

| | | | | | |
|---|---|---|---|---|---|
| −0.05 | 0.20 | −0.02 | 0.08 | −0.48 | 0.77 |

SMAX=.588963 AZMAX=210.00 RMAX=45.00 HMAX=15.00

| PLANE-A | | PLANE-C | | AXIS-P | | AXIS-B | | AXIS-T | | AXIS-Y | | AXIS-X | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP |
| 225.49 | 46.92 | 74.51 | 46.92 | 60.00 | .00 | 330.00 | 15.00 | 150.00 | 75.00 | 225.49 | 43.08 | 74.51 | 43.08 |
| 218.83 | 49.02 | 74.51 | 46.92 | 56.37 | 1.10 | 325.99 | 18.76 | 149.61 | 71.20 | 218.83 | 40.98 | 74.51 | 43.08 |
| 237.50 | 44.36 | 74.51 | 46.92 | 246.19 | 1.29 | 336.39 | 8.59 | 147.69 | 81.31 | 237.50 | 45.64 | 74.51 | 43.08 |
| 226.33 | 48.60 | 75.40 | 45.25 | 60.46 | 1.73 | 330.00 | 15.00 | 156.89 | 74.90 | 226.33 | 41.40 | 75.40 | 44.75 |
| 224.60 | 45.25 | 73.67 | 48.60 | 239.54 | 1.73 | 330.00 | 15.00 | 143.11 | 74.90 | 224.60 | 44.75 | 73.67 | 41.40 |
| 225.49 | 46.92 | 47.04 | 43.09 | 46.24 | 1.92 | 316.21 | .77 | 204.25 | 87.94 | 225.49 | 43.08 | 47.04 | 46.91 |
| 225.49 | 46.92 | 115.48 | 69.90 | 265.49 | 13.89 | 7.51 | 40.13 | 160.36 | 46.55 | 225.49 | 43.08 | 115.48 | 20.10 |

ANGLE OF ROTATION

-0.08 0.17 -0.05 0.05 -0.33 0.73

SMAX=.549894 AZMAX=168.69 RMAX=72.63 HMAX=50.13

| PLANE-A | | PLANE-C | | AXIS-P | | AXIS-B | | AXIS-T | | AXIS-Y | | AXIS-X | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP |
| 213.49 | 52.28 | 114.81 | 78.96 | 259.42 | 17.29 | 11.31 | 50.13 | 157.02 | 34.61 | 213.49 | 37.72 | 114.81 | 11.04 |
| 206.38 | 82.03 | 114.81 | 78.96 | 70.34 | 2.14 | 331.52 | 76.33 | 160.86 | 13.50 | 206.38 | 7.97 | 114.81 | 11.04 |
| 215.26 | 47.08 | 114.81 | 78.96 | 261.94 | 20.25 | 13.57 | 44.98 | 155.13 | 38.09 | 215.26 | 42.92 | 114.81 | 11.04 |
| 224.12 | 54.93 | 122.10 | 73.48 | 266.77 | 11.85 | 11.31 | 50.13 | 167.54 | 37.40 | 224.12 | 35.07 | 122.10 | 16.52 |
| 202.23 | 50.64 | 107.79 | 84.61 | 251.65 | 22.46 | 11.31 | 50.13 | 147.27 | 30.99 | 202.23 | 39.36 | 107.79 | 5.39 |
| 213.49 | 52.28 | 108.96 | 72.03 | 255.29 | 12.39 | 358.79 | 46.75 | 154.44 | 40.60 | 213.49 | 37.72 | 108.96 | 17.97 |
| 213.49 | 52.28 | 306.98 | 85.50 | 164.05 | 22.16 | 42.74 | 51.91 | 267.21 | 29.22 | 213.49 | 37.72 | 306.98 | 4.50 |

ANGLE OF ROTATION

−0.52 0.08 −0.14 0.14 −0.14 0.33

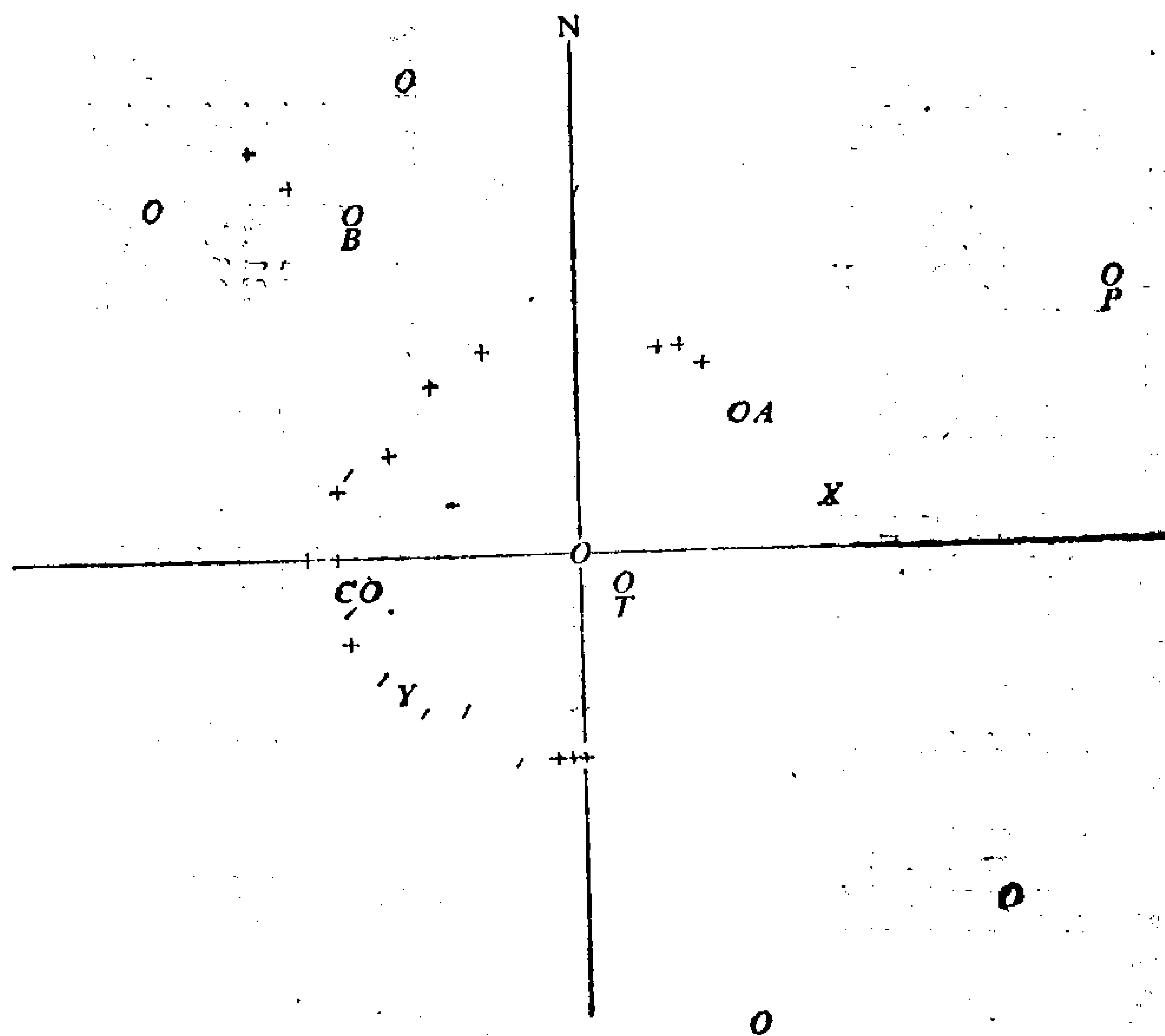


图 4-附2

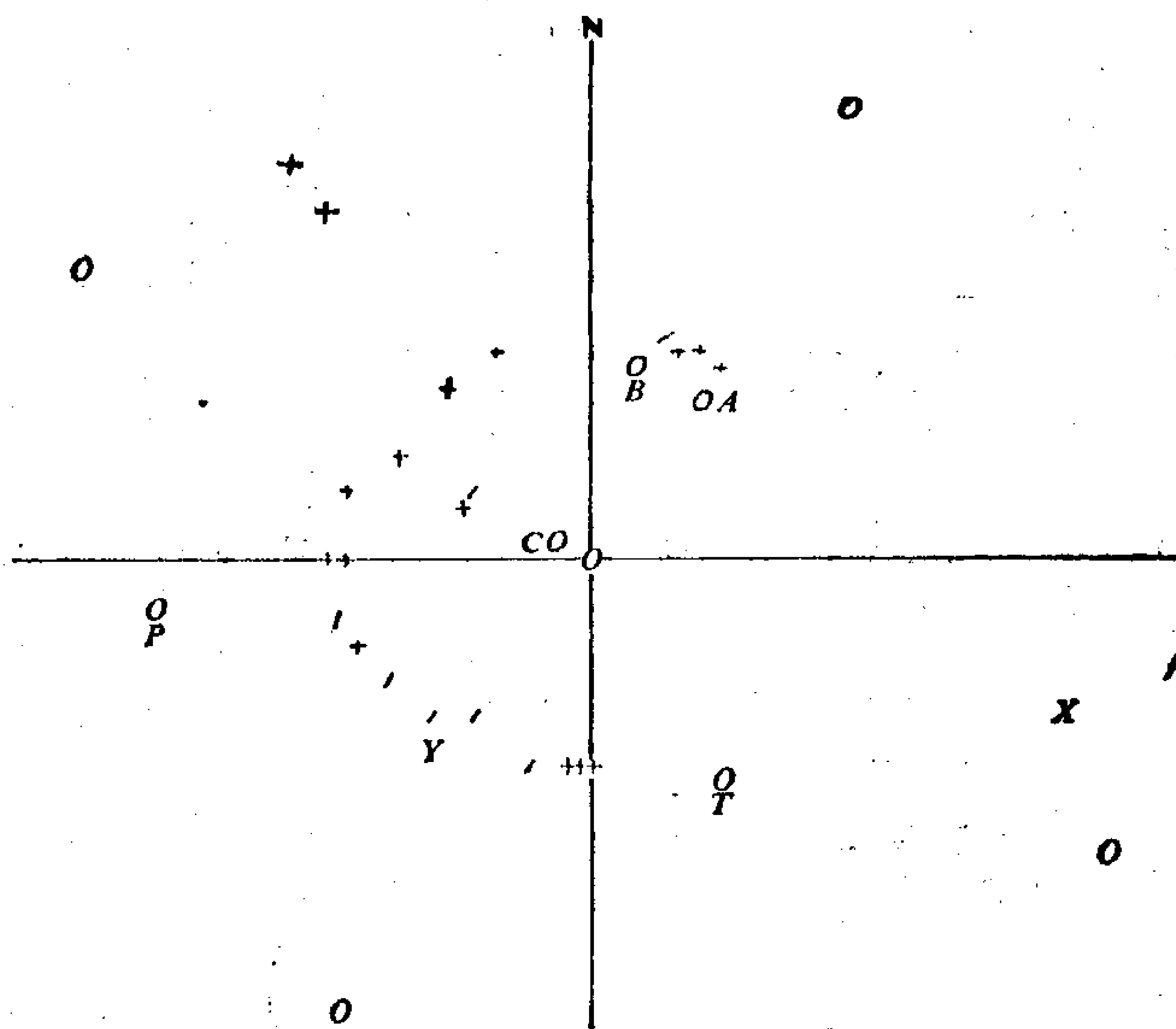


图 4-附3

SMAX=.529373 AZMAX=180.00 RMAX=60.00 HMAX=30.00

| PLANE-A | | PLANE-C | | AXIS-P | | AXIS-B | | AXIS-T | | AXIS-Y | | AXIS-X | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP |
| 229.11 | 41.41 | 106.10 | 64.34 | 262.37 | 12.95 | .00 | 30.00 | 151.81 | 56.77 | 229.11 | 48.59 | 106.10 | 25.66 |
| 224.18 | 45.58 | 106.10 | 64.34 | 259.12 | 10.88 | 356.74 | 34.61 | 154.21 | 53.23 | 224.18 | 44.42 | 106.10 | 25.66 |
| 249.92 | 30.76 | 106.10 | 64.34 | 273.16 | 17.54 | 8.29 | 15.79 | 137.82 | 66.04 | 249.92 | 59.24 | 106.10 | 25.66 |
| 234.84 | 45.08 | 109.54 | 59.91 | 265.15 | 8.33 | .00 | 30.00 | 161.26 | 58.63 | 234.84 | 44.92 | 109.54 | 30.09 |
| 206.31 | 32.78 | 97.05 | 78.00 | 253.21 | 26.58 | .00 | 30.00 | 129.64 | 47.85 | 206.31 | 57.22 | 97.05 | 12.00 |
| 229.11 | 41.41 | 59.62 | 49.07 | 234.71 | 3.85 | 325.07 | 5.23 | 108.55 | 83.50 | 229.11 | 48.59 | 59.62 | 40.93 |
| 229.11 | 41.41 | 117.07 | 71.69 | 269.96 | 17.95 | 13.36 | 35.60 | 158.23 | 48.81 | 229.11 | 48.59 | 117.07 | 18.31 |

ANGLE OF ROTATION

−0.8 0.27 −0.08 0.27 −0.73 0.20

SMAX=.507291 AZMAX=151.52 RMAX=−9.90 HMAX=48.93

| PLANE-A | | PLANE-C | | AXIS-P | | AXIS-B | | AXIS-T | | AXIS-Y | | AXIS-X | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP | AZIMUTH | DIP |
| 125.97 | 83.51 | 221.51 | 49.67 | 270.56 | 22.20 | 28.48 | 48.93 | 165.49 | 32.52 | 125.97 | 6.49 | 221.51 | 40.33 |
| 112.36 | 68.87 | 221.51 | 49.67 | 261.03 | 11.70 | 1.85 | 42.20 | 158.89 | 45.44 | 112.36 | 21.13 | 221.51 | 40.33 |
| 316.44 | 84.22 | 221.51 | 49.67 | 172.05 | 22.74 | 53.15 | 49.08 | 277.19 | 31.93 | 316.44 | 5.78 | 221.51 | 40.33 |
| 130.11 | 80.03 | 228.39 | 50.67 | 275.19 | 19.01 | 28.48 | 48.93 | 171.34 | 34.79 | 130.11 | 9.97 | 228.39 | 39.33 |
| 296.49 | 88.27 | 204.99 | 48.98 | 152.99 | 28.28 | 28.48 | 48.93 | 258.90 | 29.04 | 296.49 | 1.73 | 204.99 | 41.02 |
| 125.97 | 83.51 | 221.87 | 47.90 | 271.38 | 23.22 | 28.93 | 47.16 | 164.79 | 33.64 | 125.97 | 6.49 | 221.87 | 42.10 |
| 125.97 | 83.51 | 218.45 | 69.19 | 263.89 | 9.87 | 19.53 | 68.11 | 170.39 | 19.34 | 125.97 | 6.49 | 218.45 | 20.81 |

ANGLE OF ROTATION

−0.33 0.27 −0.08 0.23 −0.02 0.36

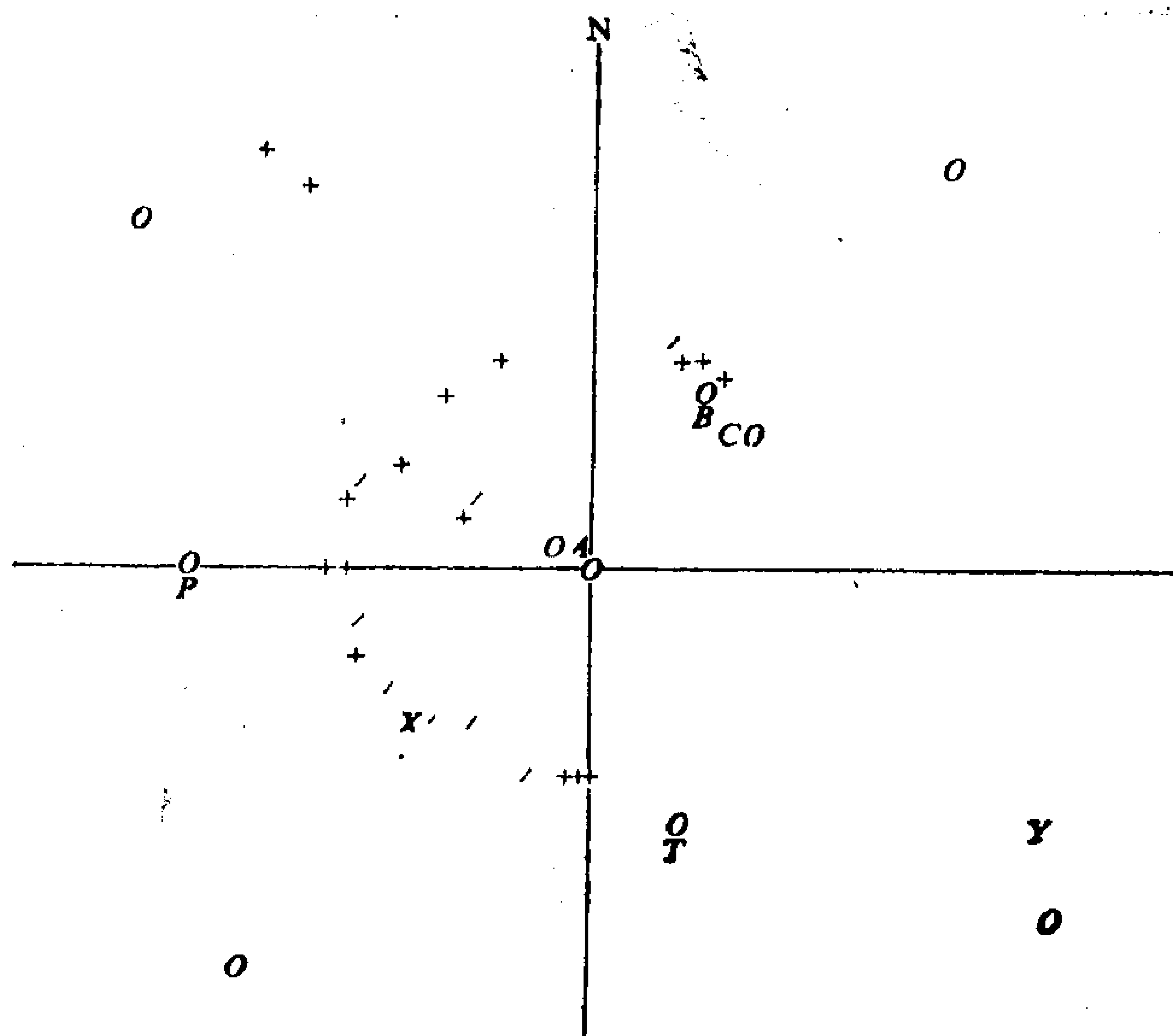



图 4-附4

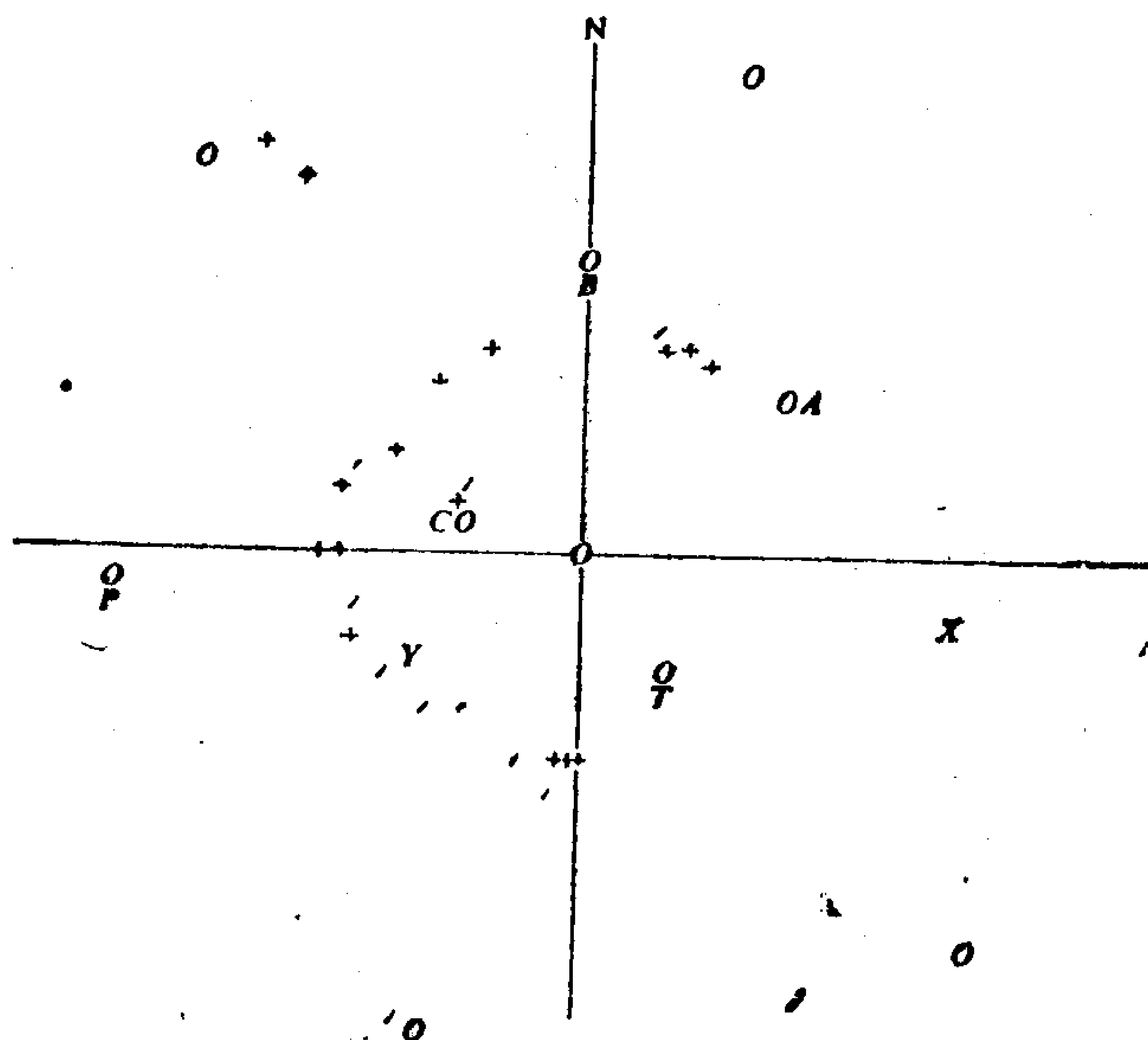



图 4-附5

下面的五张附图是利用绘图程序得到的。

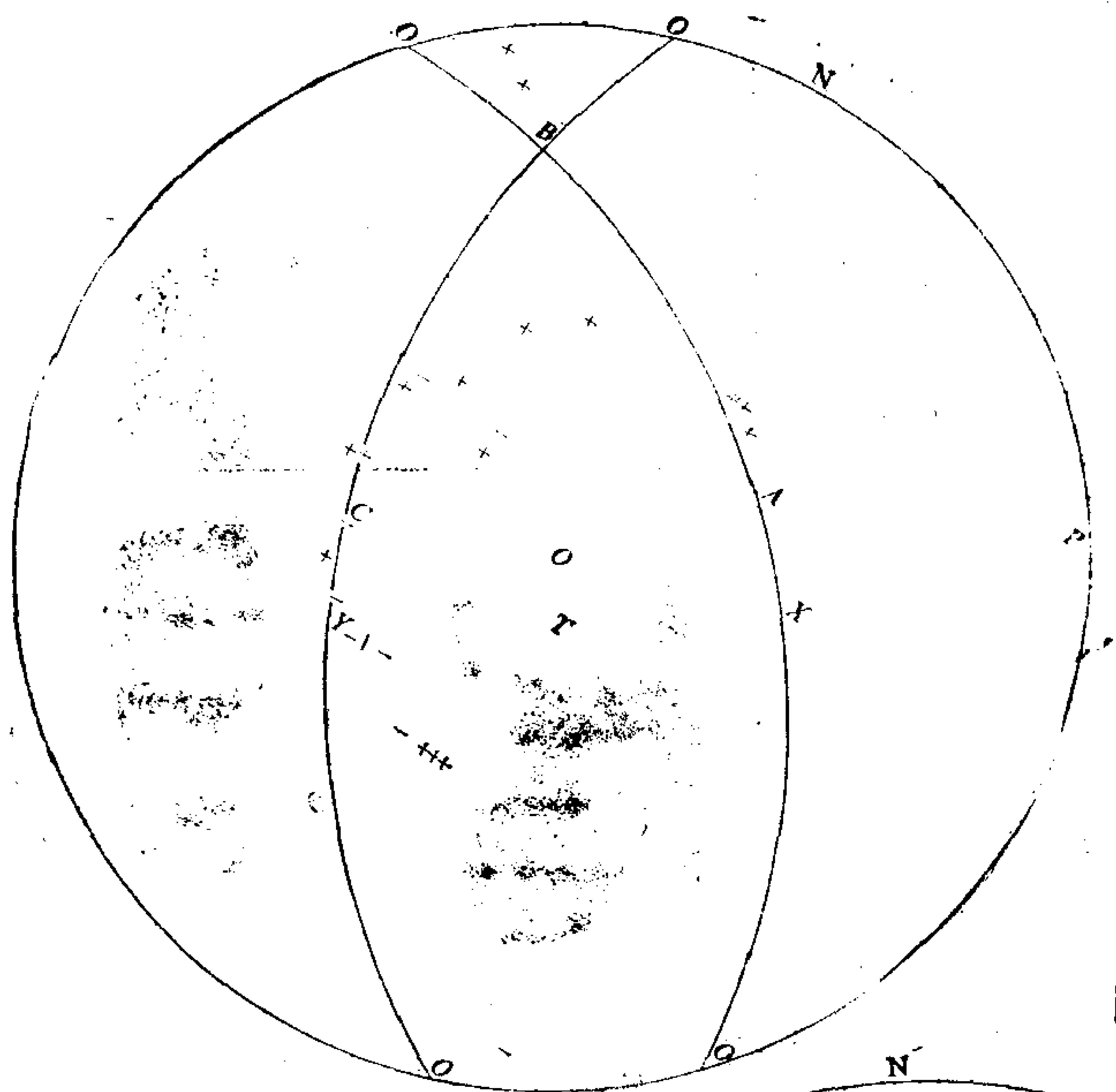



图 4-附6

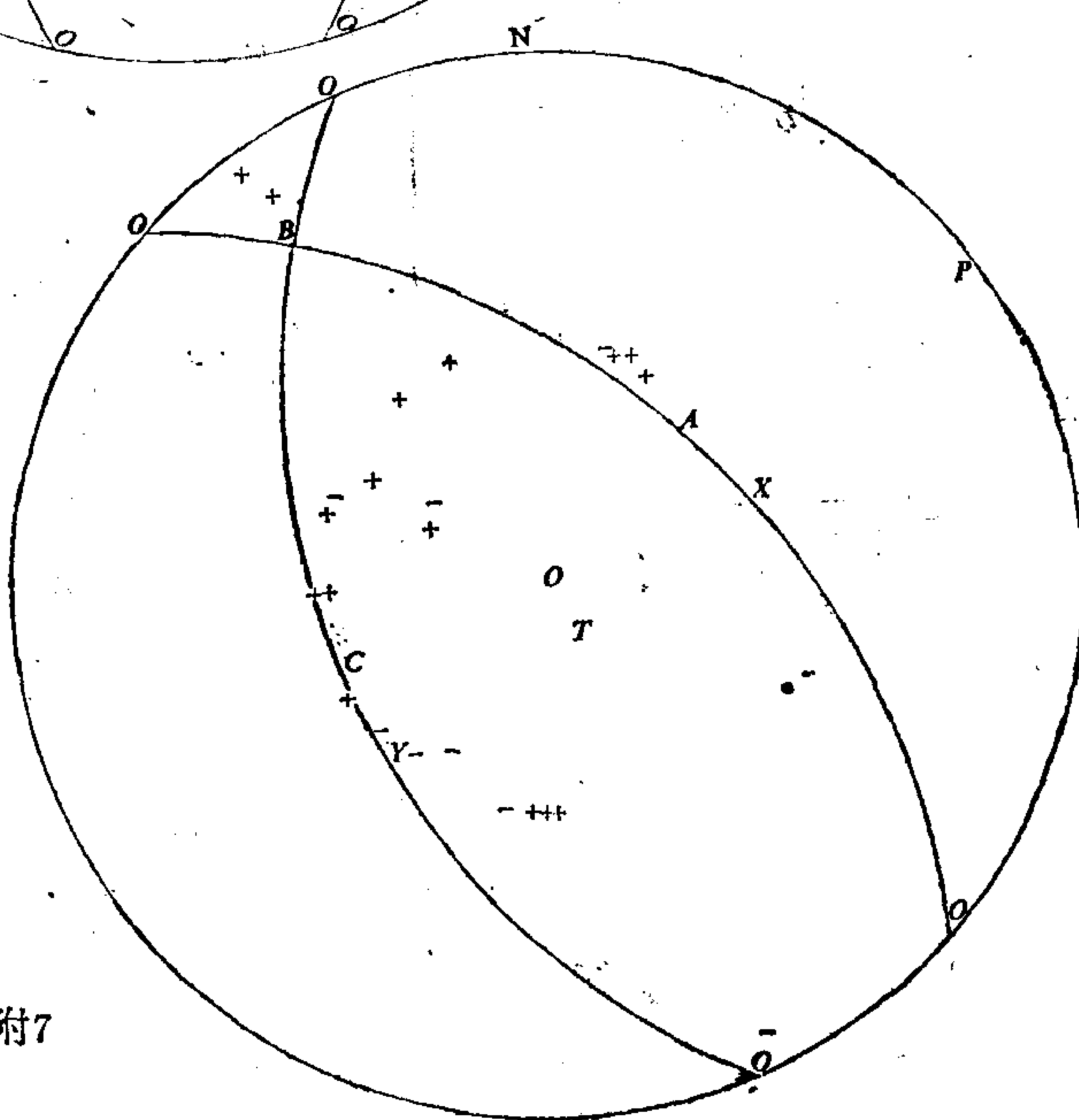



图 4-附7

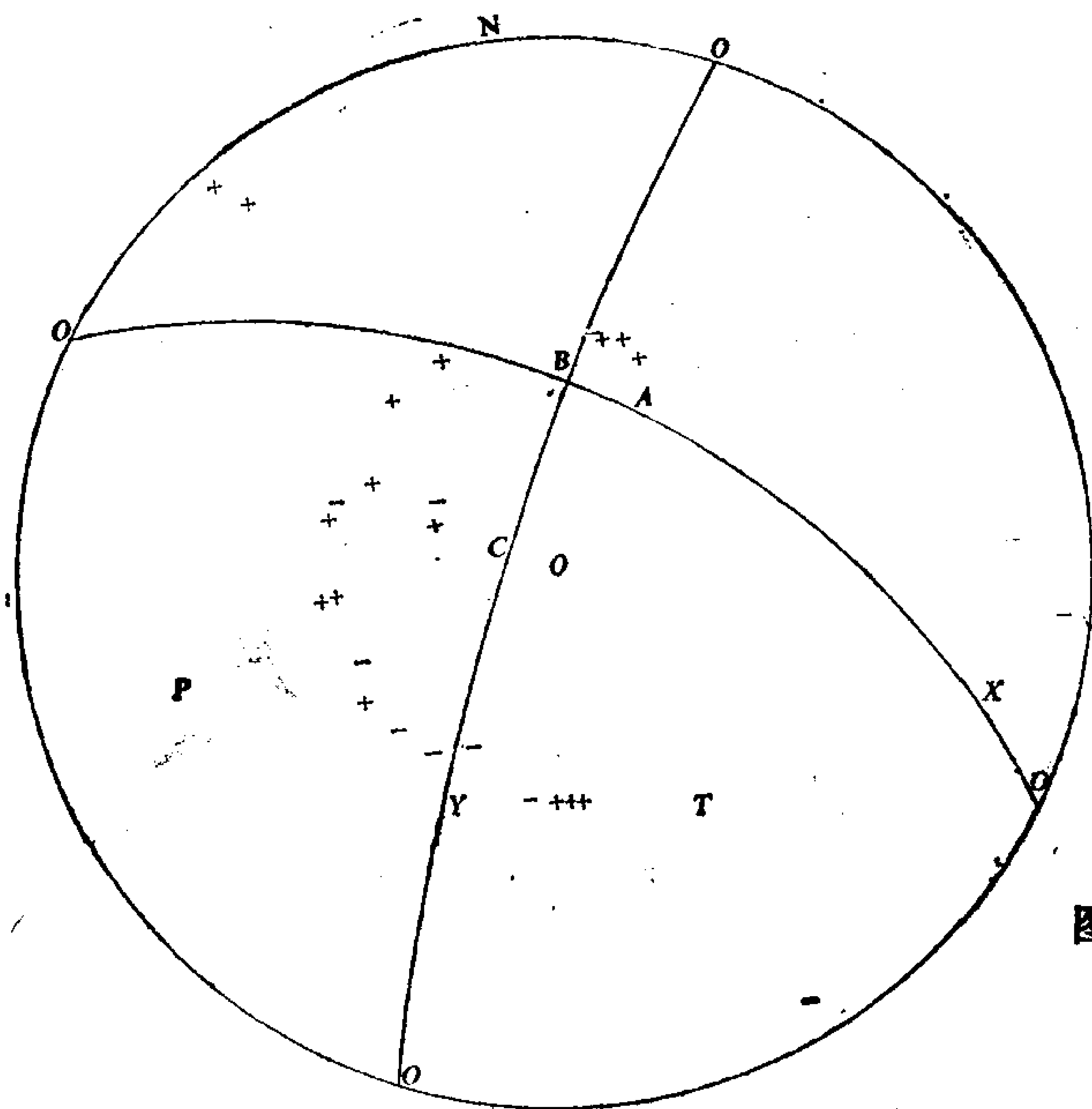



图 4-附8

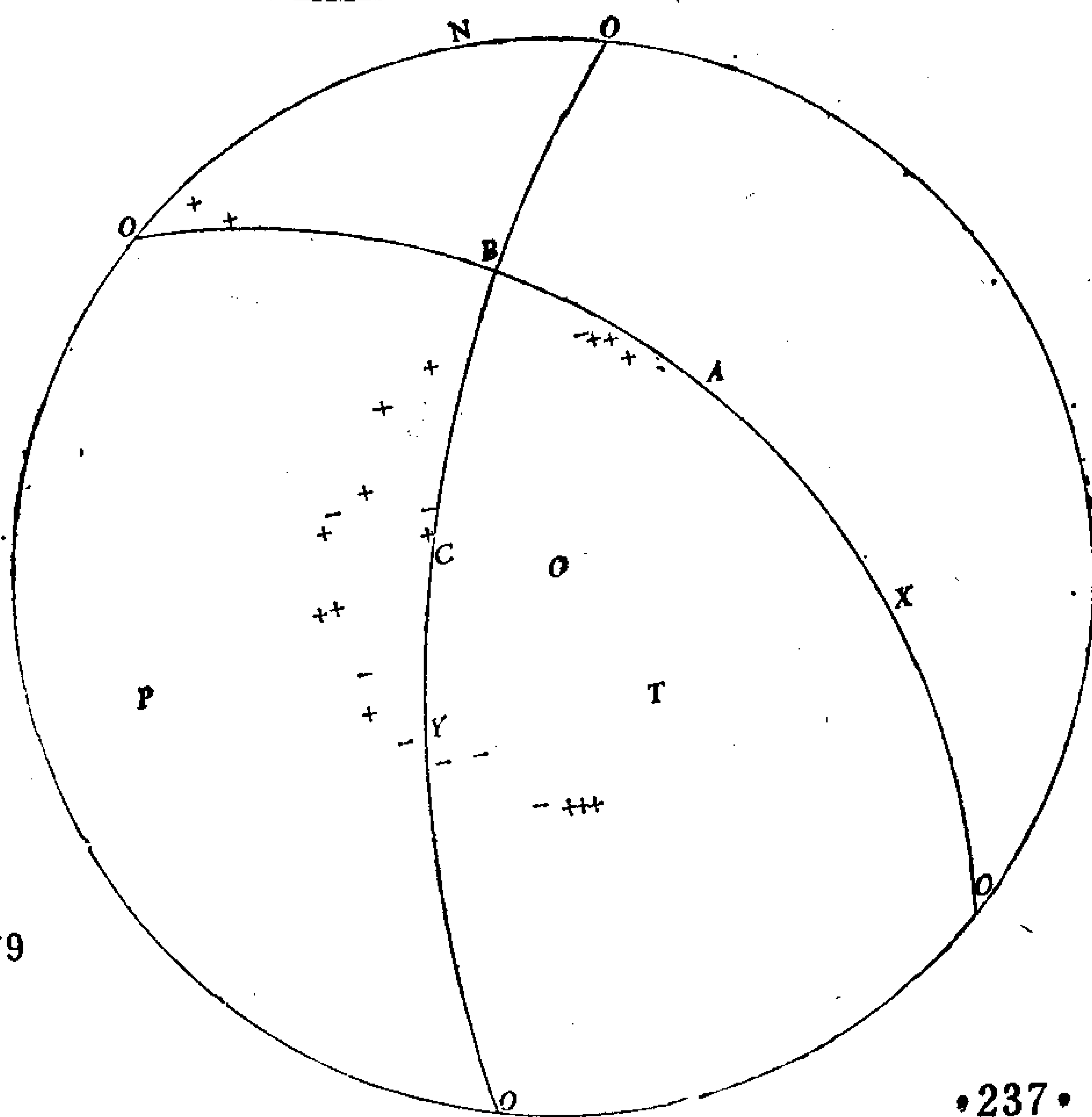



图 4-附9

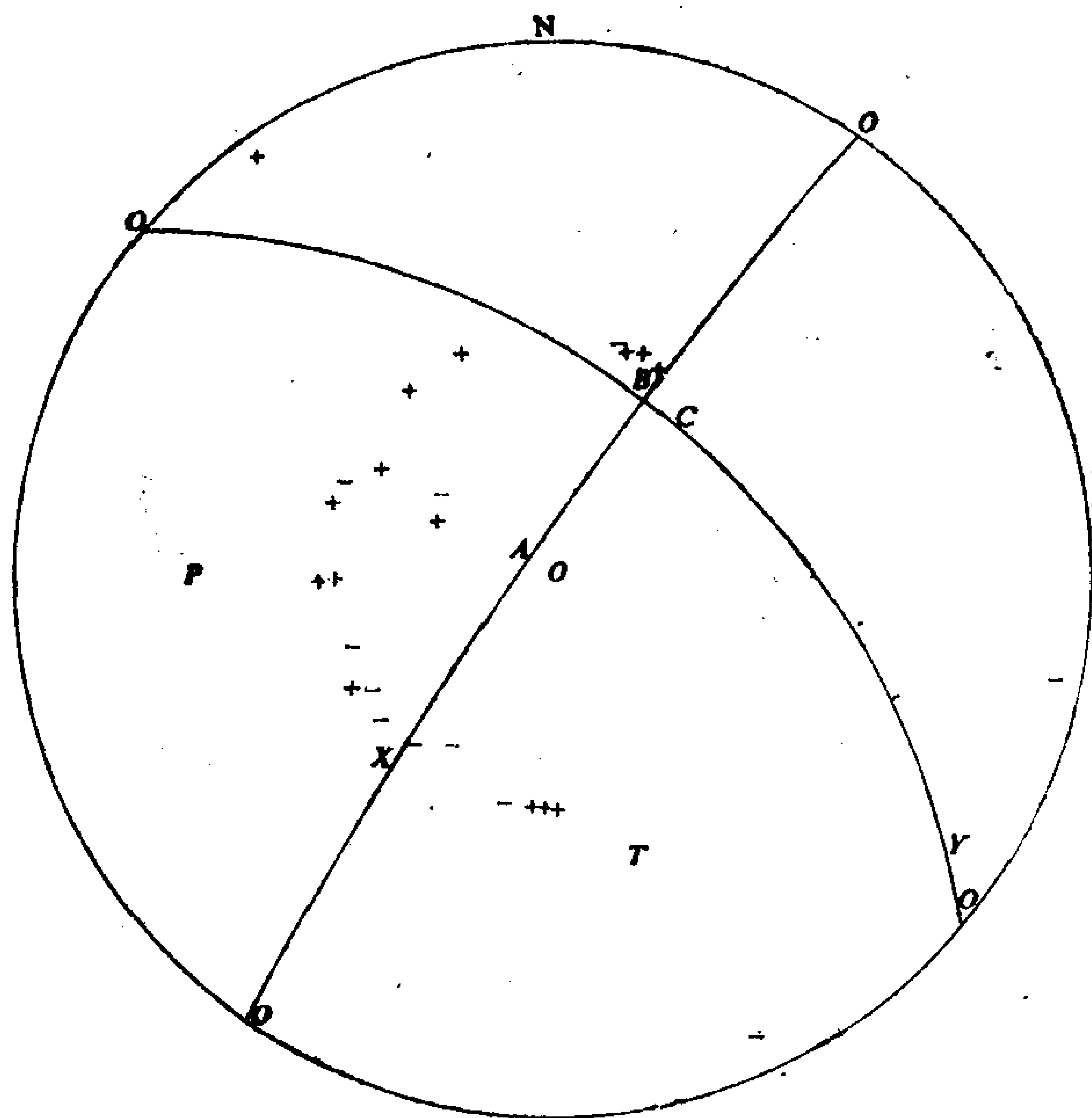



图 4-附10

# 离源角表

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2.0 | 3.0 | 4.0 | 5.0 | 6.0 | 7.0 | 8.0 | 9.0 | 10.0 | 11.0 | 12.0 | 13.0 | 14.0 | 15.0 | 16.0 |
| 17.0 | 18.0 | 19.0 | 20.0 | 20.5 | 21.0 | 21.5 | 22.0 | 22.5 | 23.0 | 23.5 | 24.0 | 24.5 | | |
| 25.0 | 25.5 | 26.0 | 26.5 | 27.0 | 27.5 | 28.0 | 28.5 | 29.0 | 29.5 | 30.0 | 30.5 | 31.0 | | |
| 31.5 | 32.0 | 32.5 | 33.0 | 33.5 | 34.0 | 34.5 | 35.0 | 35.5 | 36.0 | 36.5 | 37.0 | 37.5 | | |
| 38.0 | 38.5 | 39.0 | 39.5 | 40.0 | 40.5 | 41.0 | 41.5 | 42.0 | 42.5 | 43.0 | 43.5 | 44.0 | | |
| 44.5 | 45.0 | 45.5 | 46.0 | 46.5 | 47.0 | 47.5 | 48.0 | 48.5 | 49.0 | 49.5 | 50.0 | 50.5 | | |
| 51.0 | 51.5 | 52.0 | 52.5 | 53.0 | 53.5 | 54.0 | 54.5 | 55.0 | 55.5 | 56.0 | 56.5 | 57.0 | | |
| 57.5 | 58.0 | 58.5 | 59.0 | 59.5 | 60.0 | 60.5 | 61.0 | 61.5 | 62.0 | 62.5 | 63.0 | 63.5 | | |
| 64.0 | 64.5 | 65.0 | 65.5 | 66.0 | 66.5 | 67.0 | 67.5 | 68.0 | 68.5 | 69.0 | 69.5 | 70.0 | | |
| 70.5 | 71.0 | 71.5 | 72.0 | 72.5 | 73.0 | 73.5 | 74.0 | 74.5 | 75.0 | 75.5 | 76.0 | 76.5 | | |
| 77.0 | 77.5 | 78.0 | 78.5 | 79.0 | 79.5 | 80.0 | 80.5 | 81.0 | 81.5 | 82.0 | 82.5 | 83.0 | | |
| 83.5 | 84.0 | 84.5 | 85.0 | 85.5 | 86.0 | 86.5 | 87.0 | 87.5 | 88.0 | 88.5 | 89.0 | 89.5 | | |
| 90.0 | 90.5 | 91.0 | 91.5 | 92.0 | 92.5 | 93.0 | 93.5 | 94.0 | 94.5 | 95.0 | 95.5 | 96.0 | | |
| 96.5 | 97.0 | 97.5 | 98.0 | 98.5 | 99.0 | 99.5 | 100.0 | | | | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 47.9 | 47.8 | 47.8 | 47.7 | 47.6 | 47.5 | 47.4 | 47.3 | 47.2 |
| 47.1 | 47.0 | 46.7 | 46.2 | 45.5 | 44.4 | 43.0 | 41.5 | 39.5 |
| 37.53 | 36.62 | 35.73 | 34.86 | 34.02 | 33.22 | 32.5 | 31.85 | |
| 31.27 | 30.76 | 30.31 | 29.92 | 25.59 | 29.32 | 29.11 | 28.95 | |
| 28.85 | 28.79 | 28.74 | 28.71 | 28.67 | 28.60 | 28.50 | 28.40 | |
| 28.31 | 28.23 | 28.15 | 28.07 | 27.97 | 27.88 | 27.77 | 27.67 | |
| 27.56 | 27.44 | 27.32 | 27.20 | 27.08 | 26.96 | 26.86 | 26.73 | |
| 26.62 | 26.50 | 26.39 | 26.27 | 26.15 | 26.03 | 25.91 | 25.79 | |
| 25.67 | 25.54 | 25.42 | 25.29 | 25.16 | 25.03 | 24.91 | 24.78 | |
| 24.65 | 24.52 | 24.40 | 24.27 | 24.15 | 24.03 | 23.90 | 23.78 | |

续表

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 23.66 | 23.54 | 23.42 | 23.30 | 23.18 | 23.06 | 22.94 | 22.82 | | |
| 22.70 | 22.57 | 22.45 | 22.33 | 22.21 | 22.08 | 21.97 | 21.85 | | |
| 21.74 | 21.63 | 21.52 | 21.41 | 21.30 | 21.19 | 21.08 | 20.97 | | |
| 20.86 | 20.75 | 20.65 | 20.54 | 20.43 | 20.32 | 20.21 | 20.10 | | |
| 19.98 | 19.87 | 19.75 | 19.62 | 19.50 | 19.36 | 19.22 | 19.07 | | |
| 18.93 | 18.80 | 18.68 | 18.55 | 18.44 | 18.32 | 18.20 | 18.09 | | |
| 17.97 | 17.85 | 17.72 | 17.60 | 17.47 | 17.34 | 17.21 | 17.08 | | |
| 16.95 | 16.83 | 16.70 | 16.58 | 16.45 | 16.31 | 16.17 | 16.02 | | |
| 15.87 | 15.72 | 15.57 | 15.44 | 15.33 | 15.24 | 15.15 | 15.05 | | |
| 14.96 | 14.87 | 14.80 | 14.74 | 14.70 | 14.67 | 14.65 | 14.64 | | |
| 14.63 | 14.62 | 14.61 | 14.60 | 14.58 | 14.57 | 14.56 | 14.54 | | |
| 14.53 | 14.52 | 14.51 | 14.49 | 14.48 | 14.47 | 14.45 | 14.44 | 14.43 | |
| 56.6 | 56.6 | 56.5 | 56.4 | 56.3 | 56.2 | 56.1 | 55.9 | 55.7 | 55.5 |
| 55.4 | 55.1 | 54.9 | 53.3 | 52.0 | 51.1 | 48.2 | 45.7 | 43.15 | 42.04 |
| 40.97 | 39.93 | 38.94 | 38.00 | 37.16 | 36.40 | 35.73 | 35.13 | 34.61 | |
| 34.16 | 33.78 | 33.47 | 33.23 | 33.06 | 32.95 | 32.88 | 32.82 | 32.78 | |
| 32.73 | 32.65 | 32.54 | 32.43 | 32.33 | 32.23 | 32.13 | 32.02 | 31.91 | |
| 31.80 | 31.68 | 31.56 | 31.43 | 31.30 | 31.16 | 31.02 | 30.88 | 30.74 | |
| 30.60 | 30.46 | 30.32 | 30.19 | 30.05 | 29.91 | 29.78 | 29.64 | 29.50 | |
| 29.37 | 29.22 | 29.08 | 28.94 | 28.79 | 28.64 | 28.49 | 28.34 | 28.19 | |
| 28.04 | 27.89 | 27.74 | 27.60 | 27.45 | 27.31 | 27.17 | 27.03 | 26.89 | |
| 26.75 | 26.61 | 26.48 | 26.34 | 26.20 | 26.06 | 25.91 | 25.77 | 25.63 | |
| 25.49 | 25.35 | 25.21 | 25.07 | 24.93 | 24.80 | 24.67 | 24.54 | 24.42 | |
| 24.29 | 24.17 | 24.04 | 23.92 | 23.79 | 23.67 | 23.54 | 23.41 | 23.29 | |
| 23.16 | 23.04 | 22.91 | 22.79 | 22.66 | 22.52 | 22.38 | 22.24 | 22.09 | |
| 21.94 | 21.78 | 21.61 | 21.45 | 21.29 | 21.14 | 21.00 | 20.88 | 20.76 | |
| 20.63 | 20.49 | 20.35 | 20.20 | 20.05 | 19.90 | 19.76 | 19.62 | 19.48 | |
| 19.34 | 19.20 | 19.05 | 18.91 | 18.76 | 18.60 | 18.45 | 18.28 | 18.11 | |
| 17.94 | 17.77 | 17.61 | 17.46 | 17.34 | 17.23 | 17.13 | 17.03 | 16.93 | |
| 16.83 | 16.74 | 16.66 | 16.61 | 16.58 | 16.58 | 16.57 | 16.55 | 16.54 | |
| 16.52 | 16.51 | 16.49 | 16.48 | 16.46 | 16.45 | 16.44 | 16.42 | 16.41 | |
| 16.39 | 16.38 | 16.36 | 16.35 | 16.33 | 16.32 | | | | |

续表

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 89.0 | 89.0 | 88.0 | 86.0 | 85.0 | 85.0 | 84.0 | 83.0 | 82.0 | | |
| 81.0 | 80.0 | 79.0 | 77.0 | 74.0 | 71.0 | 67.0 | 63.0 | 59.0 | | |
| 54.21 | 52.60 | 51.09 | 49 66 | 48.31 | 47.07 | 45.96 | 44.97 | | | |
| 44.09 | 43.32 | 42.65 | 42.08 | 41.60 | 41.22 | 40.92 | 40.71 | | | |
| 40.59 | 40.51 | 40.44 | 40.39 | 40.32 | 40.21 | 40.06 | 39.90 | | | |
| 39.76 | 39.63 | 39.50 | 39.37 | 39.22 | 39.07 | 38.92 | 38.75 | | | |
| 38.58 | 38.40 | 38.22 | 38.04 | 37.85 | 37.67 | 37.49 | 37.32 | | | |
| 37.14 | 36.97 | 36.79 | 36.61 | 36.44 | 36.26 | 36.08 | 35.90 | | | |
| 35.72 | 35.53 | 35.35 | 35.16 | 34.97 | 34.77 | 34.58 | 34.39 | | | |
| 34.20 | 34.01 | 33.82 | 33.64 | 33.45 | 33.27 | 33.09 | 32.92 | | | |
| 32.74 | 32.57 | 32.40 | 32.22 | 32.04 | 31.87 | 31.69 | 31.51 | | | |
| 31.33 | 31.16 | 30.98 | 30.80 | 30.62 | 30.44 | 30.27 | 30.10 | | | |
| 29.94 | 29.78 | 29.62 | 29.47 | 29.31 | 29.16 | 29.01 | 28.86 | | | |
| 28.71 | 28.55 | 28.39 | 28.21 | 28.05 | 27.89 | 27.74 | 27.59 | | | |
| 27.43 | 27.26 | 27.09 | 26.91 | 26.72 | 26.53 | 26.33 | 26.13 | | | |
| 25.94 | 25.74 | 25.55 | 25.37 | 25.21 | 25.06 | 24.91 | 24.75 | | | |
| 24.58 | 24.41 | 24.23 | 24.05 | 23.86 | 23.68 | 23.49 | 23.31 | | | |
| 23.14 | 22.96 | 22.79 | 22.62 | 22.44 | 22.25 | 22.03 | 21.81 | | | |
| 21.60 | 21.40 | 21.22 | 21.05 | 20.89 | 20.75 | 20.61 | 20.49 | | | |
| 20.37 | 20.26 | 20.16 | 20.08 | 20.00 | 19.94 | 19.89 | 19.86 | | | |
| 19.84 | 19.84 | 19.83 | 19.82 | 19.80 | 19.78 | 19.76 | 19.74 | | | |
| 19.73 | 19.71 | 19.69 | 19.19 | 7319. | 7119. | 6919. | 6819.66 | 19.64 | 19.62 | 19.60 |
| 19.59 | 19.57 | | | | | | | | | |

# 第五章　震源的计算机定位

1960年前后，国际上兴起了地震震源计算机定位的热潮。这就是采用计算机测定地震台网记录到的地震基本参数，例如发震时刻、震中位置及震源深度等。后来，经过很多人的研究改进，计算机定位的算法及程序已得到了广泛的应用。

测定震源位置及发震时刻最常用的方法是 Geiger（1912）引入的，通常称为Geiger法。这种方法实质上是将非线性问题化为线性问题，然后利用最小二乘原理迭代求解。通过反复修定，以得到震源参数的最佳值。

近年来，由于数学方法的发展，使计算机定位问题有了更普遍而行之有效的方法。利用广义逆或最优化方法修定震源参数，可以得到更快的收敛速度和更高的精度。

一般说来，计算机定位要确定四个参数，即震源的空间坐标($\lambda$，$\varphi$，$h$)及发震时刻($t$)。为此，我们需要知道台网各个台站的坐标，以及至少四个台的地震波到时。由台网提供的到时数据主要是指$P$ 波的到时，因为各种续至波的相位通常是难以识别的。如果地震发生在台网内，那么只利用$P$ 波到时定位并不困难。如果地震发生在台网以外，这种解答是不确定的。为了进行参数的修定，一般要用到走时表(有时是区域走时表)或对地壳平均速度作出假设。

这一章，我们将详细讨论近震及远震的计算机定位方法。特别是各种修定震中的算法。最后给出实用的定位程序及计算实例。

# §5-1 近震的初步定位

利用区域台网记录对近震及地方震定位，可以得到较高的定位精度。定位计算一般分两步进行，即初定和修定。初定震中时，仅须根据区域台网提供的直达波到时，将走时方程线性化，从而解出四个基本震源参数。在初定震源参数的基础上，然后采用“中国地区近震走时表”（或区域走时表），以最小二乘法或其它最优化方法对这些参数进行修定。为了保证修定的质量，必须在修定前对台站数据进行检查，剔除那些明显错误的数据。

这一节，讨论初定震源参数及检查台站数据的算法。

## （一）坐标的变换

对于近震及地方震的计算，采用平面坐标比球面坐标方便。为此需要将经纬度换算为局部的直角坐标系。

这里需要指出，为了处理数据方便，一般是将国内基准台网及区域台网的全部台站的经纬度及高程均输入计算机内。但是，在作具体计算时，只用到其中一部分台站，也就是只需要把这部分台站的经纬度（球面坐标）化为平面坐标系。

因而，直角坐标系的原点是浮动的，每次都选来尽可能的靠近震中区，以减少变换过程带来的误差。在计算程序中，坐标原点确定在接收到该次地震信号各台站经纬度的平均值$\lambda_0$及$\varphi_0$上，$x$轴向东，$y$轴向北，如图5-1-1所示。

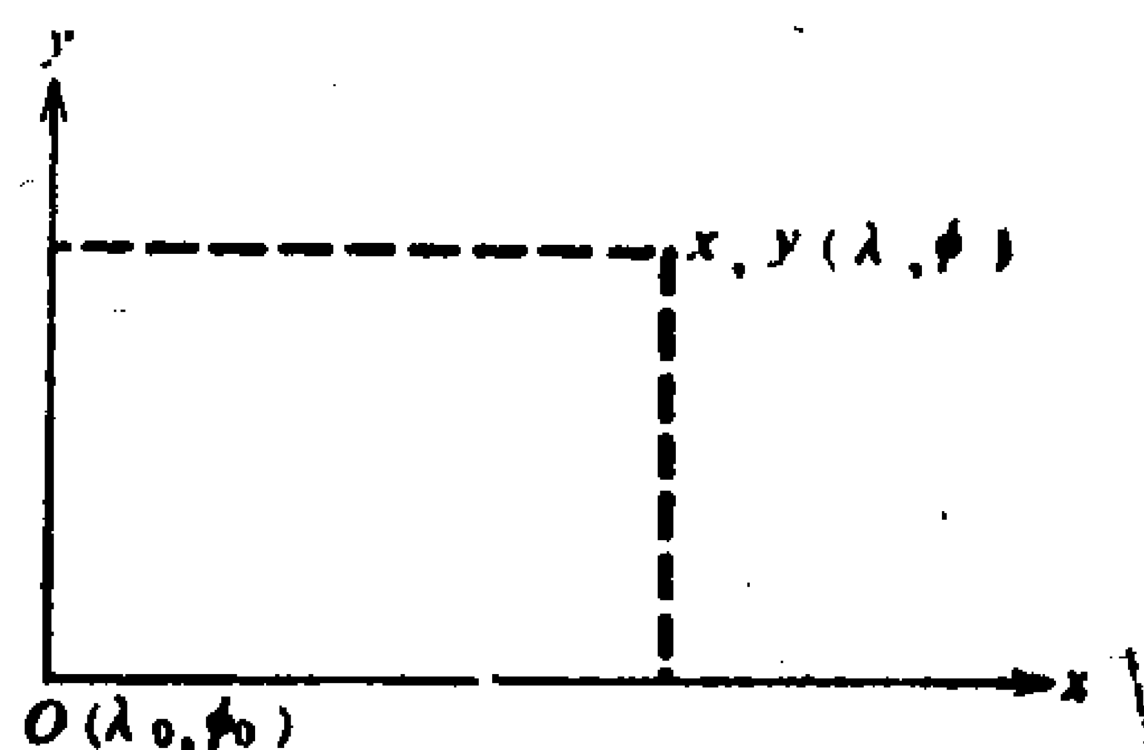



图 5-1-1

于是，地面上任一点（其经纬度为$\lambda$、$\varphi$）相对于坐标原点$O$（$\lambda_0$，$\varphi_0$）的直角坐标$x$、$y$的换算公式为

$$x = R_n(\lambda - \lambda_0)\cos\varphi_0,$$
$$y = R_m(\phi - \phi_0) + \frac{x^2\tan\varphi_0}{2R_n}. \qquad (5\text{-}1.1)$$

其中参数

$$R_n = \frac{a}{(1 - e^2\sin^2\phi_0)^{1/2}},$$

$$R_m = \frac{a(1 - e^2)}{(1 - e^2\sin^2\phi_0)^{3/2}}.$$

而参数$R_n$及$R_m$中的常数又为

$$e^2 = \frac{a^2 - b^2}{a^2} = 0.0066935165,\text{（地球的偏心率）}$$

$a = 6378.160\text{km}$。　　　　（地球长半径）

$b = 6356.778\text{km}$。　　　　（地球短半径）

经换算为平面坐标系后，就可以用计算机确定震源参数。但这样得到的震中位置仍是局部的$x$、$y$坐标。因此，必须化为经纬度$\lambda$、$\phi$输出。将直角坐标值$x$、$y$化为经纬度$\lambda$，$\varphi$的换算公式为

$$\lambda = \frac{x}{R_n\cos\phi_0} + \lambda_0,$$

$$\phi = \frac{y}{R_m} - \frac{x^2\tan\phi_0}{2R_mR_n} + \phi_0. \qquad (5\text{-}1.2)$$

## （二）初定震源参数

为了使修定震源参数的初始值尽可能接近实际值，有必要对震源参数进行初定，以减少修定时的迭代次数。实际计算表明，只要输入数据正确，初定的震源位置及发震时刻足够精确，就可供地震速报之用。其具体方法如下。

设接收到某次地震直达波到时的台站有$n$个($n \geq 4$)。其中任一台站的坐标为($x_i$，$y_i$)，相应的直达波到时为$T_i$。待求的震

中位置为 $x$，$y$，震源深度为 $z$，发震时刻为 $T$。按照均匀地壳模型，可列出下列方程：

$$(x-x_i)^2+(y-y_i)^2+z^2=V^2(T_i-T)^2,$$

$$i=1,\ 2,\cdots,\ n. \qquad (5\text{-}1.3)$$

式中 $V$ 为地震波在地壳中的平均传播速度，因区域不同而异。例如，北京台网取 $V=6.0\text{km/s}$，四川台网取 $V=5.5\text{km/s}$，均能取得较好的计算结果。

式(5-1.3)为非线性方程组，为便于直接求解，首先应将其线性化，变为线性方程组。为此将(5-1.3)式展开得

$$x^2-2x_ix+x_i^2+y^2-2y_iy+y_i^2+z^2$$
$$=V^2(T_i^2-2TT_i+T^2),$$
$$i=1,\ 2,\ \cdots,\ n. \qquad (5\text{-}1.4)$$

对于第一个台，($i=1$) 可以写出

$$x^2-2x_1x+x_1^2+y^2-2y_1y+y_1^2+z^2$$
$$=V^2(T_1^2-2TT_1+T^2). \qquad (5\text{-}1.5)$$

将(5-1.4)式中 $i$ 分别代以 2，3，…，$n$ 与(5-1.5)式相减得

$$(x_i-x_1)x+(y_i-y_1)y-V^2(T_i-T_1)T$$
$$=[(x_i^2+y_i^2-V^2T_i^2)-(x_1^2+y_1^2-V^2T_1^2)]/2,$$
$$i=2,\ 3,\ \cdots,\ n. \qquad (5\text{-}1.6)$$

共有 $n-1$ 个线性方程式。(5-1.6)式已消去了深度参数 $z$，只有 $x$、$y$、$T$ 三个未知数。由于 $n-1$ 一般都大于 3，故(5-1.6)式所表示的线性方程的个数多于未知数个数，即构成下列所谓超定方程组

$$(x_2-x_1)x+(y_2-y_1)y-V^2(T_2-T_1)T$$
$$=[(x_2^2+y_2^2-V^2T_2^2)-(x_1^2+y_1^2-V^2T_1^2)]/2,$$
$$(x_3-x_1)x+(y_3-y_1)y-V^2(T_3-T_1)T$$
$$=[(x_3^2+y_3^2-V^2T_3^2)-(x_1^2+y_1^2-V^2T_1^2)]/2, \qquad (5\text{-}1.7)$$
$$\cdots\cdots,$$
$$(x_n-x_1)x+(y_n-y_1)y-V^2(T_n-T_1)T$$

$$=[(x_n^2+y_n^2-V^2T_n^2)-(x_1^2+y_1^2-V^2T_1^2)]/2.$$

为了表达简洁起见，将此超定方程组用矩阵形式表达

$$Ax=b. \tag{5-1.8}$$

其中

$$A=\begin{pmatrix} x_2-x_1 & y_2-y_1 & -V^2(T_2-T_1) \\ x_3-x_1 & y_3-y_1 & -V^2(T_3-T_1) \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ x_n-x_1 & y_n-y_1 & -V^2(T_n-T_1) \end{pmatrix},$$

$$x=\begin{pmatrix} x \\ y \\ T \end{pmatrix},$$

$$b=\begin{pmatrix} [(x_2^2+y_2^2-V^2T_2^2)-(x_1^2+y_1^2-V^2T_1^2)]/2 \\ [(x_3^2+y_3^2-V^2T_3^2)-(x_1^2+y_1^2-V^2T_1^2)]/2 \\ \cdots \qquad \cdots \qquad \cdots \\ \cdots \qquad \cdots \qquad \cdots \\ [(x_n^2+y_n^2-V^2T_n^2)-(x_1^2+y_1^2-V^2T_1^2)]/2 \end{pmatrix}.$$

可以证明，矩阵 $A$ 是列满秩的。按照广义逆理论存在左逆，$A_L^{-1}=(A^TA)^{-1}A^T$，线性方程组(5-1.8)式的解答为

$$x=A_L^{-1}b=(A^TA)^{-1}A^Tb. \tag{5-1.9}$$

由此便确定了$x=(x,\ y,\ T)^T$三个震源参数。

为了定出震源深度 $z$，可把求得的震源参数$x$、$y$、$T$ 回代入(5-1.3)式，由$n$ 个台站的数据计算平均的震源深度 $z$

$$Z=\left\{\sum_{i=1}^{n}[V^2(T_i-T)^2-(x-x_i)^2-(y-y_i)^2]^{1/2}\right\}/n. \tag{5-1.10}$$

### （三）台站数据的检查

在初定震源参数$x$、$y$、$z$、$T$后，即可利用这些数值去检验由于时间服务不准或其它各种原因导致的地震波到时的错误数据。在处理大量资料时，输入一些个别的错误数据是难免的。为了保证被修定的震源参数的质量，必须对台站的资料进行检查。这可以由以下方法实现。

根据初定的震中位置$x$、$y$及各台站的坐标$x_i$、$y_i$，可以算出相对于各台站的震中距

$$d_i=\sqrt{(x_i-x)^2+(y_i-y)^2}\ .$$

由震中距$d_i$根据“中国地区近震走时表”（或区域走时表）内插求得理论走时$\Delta T_i^{理}$，则各相应的理论到时为

$$T_i^{理}=T+\Delta T_i^{理}.$$

其中$T$为发震时刻。将计算得到的理论值$T_i^{理}$与实际观测的到时$T_i^{观}$相比较，若二者相差在±（1—2）秒范围内（应根据台网时间服务实际精度而定），则认为是正常偏差。若二者相差超过此范围，则认为该台站观测到时$T_i^{观}$的数据有错误而予以剔除。

当一组数据经剔除后剩下的台站不足4个时，则停止运算。若等于或多于4个，则利用这些满足要求的台站到时继续进行震源参数的修定。

## §5-2 Geiger法修定近震震源参数

为了得出更精确的定位值，必须对初定值进行修定。Geiger法包含非线性方程线性化和最小二乘两个基本概念。下面详细讨论这种算法。

设第$i$个台直达波到时的观测值为$T_i^{观}$，对于均匀地壳模型，理

论到时值为

$$\tau_i(x,y,z,T)=\frac{1}{V}\sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}+T. \tag{5-2.1}$$

式中$x$，$y$，$z$，$T$为震源参数，$x_i$，$y_i$，$z_i$为第$i$个台的平面坐标位置及高程。$\tau_i$是$x$，$y$，$z$，$T$的函数。

我们将初定的震源参数记为$(x_0, y_0, z_0, T_0)^T$，则由此可求得相应的理论到时为

$$\tau_i(x_0, y_0, z_0, T_0)=\frac{1}{V}\sqrt{(x_0-x_i)^2+(y_0-y_i)^2+(z_0-z_i)^2}+T_0, \quad i=1, 2, \cdots, n. \tag{5-2.2}$$

如果以$(x, y, z, T)^T$表示震源参数 的真实解，且假定初定的震源系数$(x_0, y_0, z_0, T_0)^T$与真解足够接近，根据 Taylor 展开式可以得到下述近似表达式：

$$\begin{aligned}\tau_i(x, y, z, T)=&\tau_i(x_0, y_0, z_0, T_0)\\&+\left.\frac{\partial\tau_i}{\partial x}\right|_{(x_0,y_0,z_0,T_0)}\delta x+\left.\frac{\partial\tau_i}{\partial y}\right|_{(x_0,y_0,z_0,T_0)}\delta y\\&+\left.\frac{\partial\tau_i}{\partial z}\right|_{(x_0,y^0,z_0,T_0)}\delta z+\left.\frac{\partial\tau_i}{\partial T}\right|_{(x_0,y_0,z_0,T_0)}\delta T+e_i,\\&i=1, 2, \cdots, n.\end{aligned} \tag{5-2.3}$$

式中$\tau_i(x, y, z, T)$是真实到时，实际上可以认为 它 是 第$i$个台的观测到时$T_i^{观}$；$\delta x=x-x_0$，$\delta y=y-y_0$，$\delta z=z-z_0$，$\delta T=T-T_0$：$e_i$为二次及二次以上高次$T_i$截断误差项。

我们把观测到时$T_i^{观}$〔无误差时，它应等于理论值$\tau_i(x, y, z, T)$〕与由初定震源参数计算的理论到时$\tau_i(x_0, y_0, z_0, T_0)$之差称为到时残差，记为

$$R_i=\tau_i(x,\ y,\ z,\ T)-\tau_i(x_0,\ y_0,\ z_0,\ T_0)$$
$$=\left.\frac{\partial\tau_i}{\partial x}\right|_{(x_0,y_0,z_0,T_0)}\delta x+\left.\frac{\partial\tau_i}{\partial y}\right|_{(x_0,y_0,z_0,T_0)}\delta y+\left.\frac{\partial\tau_i}{\partial z}\right|_{(x_0,y_0,z_0,T_0)}\delta z+\left.\frac{\partial\tau_i}{\partial T}\right|_{(x_0,y_0,z_0,T_0)}\delta T+e_i,\qquad i=1,\ 2,\ \cdots,\ n. \tag{5-2.4}$$

根据(5-5.1)式，不难求得各偏微分项为

$$\left.\frac{\partial\tau_i}{\partial x}\right|_{(x_0,y_0,z_0,T_0)}=\frac{x_0-x_i}{V\sqrt{(x_0-x_i)^2+(y_0-y_i)^2+(z_0-z_i)^2}}=a_i,$$
$$\left.\frac{\partial\tau_i}{\partial y}\right|_{(x_0,y_0,z_0,T_0)}=\frac{y_0-y_i}{V\sqrt{(x_0-x_i)^2+(y_0-y_i)^2+(z_0-z_i)^2}}=b_i, \tag{5-2.5}$$
$$\left.\frac{\partial\tau_i}{\partial z}\right|_{(x_0,y_0,z_0,T_0)}=\frac{z_0-z_i}{V\sqrt{(x_0-x_i)^2+(y_0-y_i)^2+(z_0-z_i)^2}}=c_i,$$
$$\left.\frac{\partial\tau_i}{\partial T}\right|_{(x_0,y_0,z_0,T_0)}=1.$$

式(5-2.5)都是常系数。这样便得到一个线性方程组

$$R_i=\delta T+a_i\delta x+b_i\delta y+c_i\delta z+e_i,\quad i=1,\ 2,\ \cdots,\ n.$$

移项便得

$$e_i=R_i-\delta T-a_i\delta x-b_i\delta y-c_i\delta z,\quad i=1,\ 2,\ \cdots,\ n. \tag{5-2.6}$$

上面我们指出$e_i$是截断误差，但实际上还存在观测误差，因此把$e_i$看成包含观测误差及其它所有误差项更合适一些。

根据最小二乘原理，取误差项平方和为最小

$$\sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left[R_i - (\delta T + a_i\delta x + b_i\delta y + c_i\delta z)\right]^2 = \min. \tag{5-2.7}$$

由此条件，可分别求偏导数，并令结果为零

$$\begin{aligned}
&\frac{\partial}{\partial(\delta T)}\left(\sum_{i=1}^{n} e_i^2\right) = 0,\\
&\frac{\partial}{\partial(\delta x)}\left(\sum_{i=1}^{n} e_i^2\right) = 0,\\
&\frac{\partial}{\partial(\delta y)}\left(\sum_{i=1}^{n} e_i^2\right) = 0,\\
&\frac{\partial}{\partial(\delta z)}\left(\sum_{i=1}^{n} e_i^2\right) = 0.
\end{aligned} \tag{5-2.8}$$

不难得到下列线性方程组

$$\begin{aligned}
&n\delta T + \sum_{i=1}^{n} a_i\delta x + \sum_{i=1}^{n} b_i\delta y + \sum_{i=1}^{n} c_i\delta z = \sum_{i=1}^{n} R_i,\\
&\sum_{i=1}^{n} a_i\delta T + \sum_{i=1}^{n} a_i^2\delta x + \sum_{i=1}^{n} a_ib_i\delta y + \sum_{i=1}^{n} a_ic_i\delta z = \sum_{i=1}^{n} a_iR_i,\\
&\sum_{i=1}^{n} b_i\delta T + \sum_{i=1}^{n} a_ib_i\delta x + \sum_{i=1}^{n} b_i^2\delta y + \sum_{i=1}^{n} b_ic_i\delta z = \sum_{i=1}^{n} b_iR_i,\\
&\sum_{i=1}^{n} c_i\delta T + \sum_{i=1}^{n} a_ic_i\delta x + \sum_{i=1}^{n} b_ic_i\delta y + \sum_{i=1}^{n} c_i^2\delta z = \sum_{i=1}^{n} c_iR_i.
\end{aligned} \tag{5-2.9}$$

注意，在迭代过程中，式(5-2.9)中各系数可以写为

$$a_i = \frac{x - x_i}{VD_i},$$

$$b_i = \frac{y - y_i}{VD_i},$$

$$c_i = \frac{z - z_i}{VD_i},$$

$$R_i = (T_i - T) - \frac{D_i}{V},$$

且

$$D_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}.$$

式中$x$，$y$，$z$，$T$为迭代过程中不断改变的震源参数值。

一般说来，解方程组（5-2.9）即可求得震源参数各修定项$\delta x$，$\delta y$，$\delta z$及$\delta T$。因此，用Geiger法修定参数是一个简单易行的方法。特别是用计算机处理更为便利。但是，解方程组（5-2.9)并不一定都能得到预定的结果。这常常是因为台站分布不合理及地壳速度模型不正确而导致方程组(5-2.9)中的系数矩阵是病态的。由于求解校正量$\delta x$，$\delta y$，$\delta z$及$\delta T$的困难，故在修定震源参数过程中采用迭代法，对未知量进行分离，先求出$\delta x$，$\delta y$及$\delta z$的值，然后再回代求$\delta T$的值。每一次迭代得到的校正后的参数，都作为下一次计算的起点。

由方程组(5-2.9)的第一式得

$$\delta T = \frac{1}{n}\sum_{i=1}^{n} R_i - \frac{1}{n}\left(\sum_{i=1}^{n} a_i\delta x + \sum_{i=1}^{n} b_i\delta y + \sum_{i=1}^{n} c_i\delta z\right). \tag{5-2.10}$$

将上式代入(5-2.9)式中后面三个式子，可以得到以下线性方程组：

$$\left[\sum_{i=1}^{n} a_i^2 - \frac{\left(\sum_{i=1}^{n} a_i\right)^2}{n}\right]\delta x + \left[\sum_{i=1}^{n} a_i b_i - \frac{\left(\sum_{i=1}^{n} a_i\right)\left(\sum_{i=1}^{n} b_i\right)}{n}\right]\delta y$$

$$+\left[\sum_{i=1}^{n} a_i c_i - \frac{\left(\sum_{i=1}^{n} a_i\right)\left(\sum_{i=1}^{n} c_i\right)}{n}\right]\delta z = \sum_{i=1}^{n} a_i R_i$$

$$-\frac{\left(\sum_{i=1}^{n} a_i\right)\left(\sum_{i=1}^{n} R_i\right)}{n},$$

$$\left[\sum_{i=1}^{n} a_i b_i - \frac{\left(\sum_{i=1}^{n} a_i\right)\left(\sum_{i=1}^{n} b_i\right)}{n}\right]\delta x + \left[\sum_{i=1}^{n} b_i^2 - \frac{\left(\sum_{i=1}^{n} b_i\right)^2}{n}\right]\delta y$$

$$+\left[\sum_{i=1}^{n} b_i c_i - \frac{\left(\sum_{i=1}^{n} b_i\right)\left(\sum_{i=1}^{n} c_i\right)}{n}\right]\delta z = \sum_{i=1}^{n} b_i R_i$$

$$-\frac{\left(\sum_{i=1}^{n} b_i\right)\left(\sum_{i=1}^{n} R_i\right)}{n},$$

$$\left[\sum_{i=1}^{n} a_i c_i - \frac{\left(\sum_{i=1}^{n} a_i\right)\left(\sum_{i=1}^{n} c_i\right)}{n}\right]\delta x$$

$$+\left[\sum_{i=1}^{n} b_i c_i - \frac{\left(\sum_{i=1}^{n} b_i\right)\left(\sum_{i=1}^{n} c_i\right)}{n}\right]\delta y$$

$$+\left[\sum_{i=1}^{n} c_i^2 - \frac{\left(\sum_{i=1}^{n} c_i\right)^2}{n}\right]\delta z = \sum_{i=1}^{n} c_i R_i - \frac{\left(\sum_{i=1}^{n} c_i\right)\left(\sum_{i=1}^{n} R_i\right)}{n}.$$

(5-2.11)

方程组（5-2.11）中只有三个未知数，$\delta x$，$\delta y$，$\delta z$，利用一般的消元法即可解出这个修正项。而时间修正项 $\delta T$ 可将求出之 $\delta x$，$\delta y$，$\delta z$ 回代入(5-2.10)式中即得。

不过，用这样的方法修定震源深度$z$是不准确的。所以，我们将用试探法求震源深度(方法见后)。修定后的震中坐标值及发震时刻为

$$x' = x + \delta x,$$

$$y' = y + \delta y,$$

$$T' = T + \delta T.$$

修定项$\delta x$，$\delta y$，$\delta T$的绝对值可用来估计精度。如不满足预定的精度要求，可将修定值$x'$，$y'$及$T'$作为下次修定时的初值，再进行修定。如此迭代计算，直至满足预定精度为止。

最后应当指出，用计算机对震源参数进行计算，很大程度上依赖于输入数据的质量。因此，必须仔细地准备输入数据。由于各台的时间服务及记录质量的差别，可以用加权最小二乘法来合

理的处理数据，提取有用信息。

## §5-3 远震震源参数的修定

这一节，讨论用Geiger法对远震震源参数进行修定的算法。由于远震是在球面坐标上定位的，所以它的计算方法与近震有很大不同。

对远震震源参数进行修定时，必须给出震中经纬度及发震时刻的初始值。这个初始值可由于手算或其它任意方法估计。例如，常常取最小走时（即最靠近震中）的台站位置作为初始震中。计算时一般都采用$JB$表。目前，已编制了“中国地区$P$波走时表”。因此，对国内或邻近边境的地震，用这种走时表也可取得较好的效果。下面讨论修定震中的计算方法。

### （一）高度校正及地球椭圆扁率校正

在修定震源参数前，必须对观测走时进行台站高度校正和地球椭圆扁率校正，以便使观测结果划归到同一基准面。

台站高度校正公式如下

$$\delta_{t_h}=\frac{z_S\sqrt{1-\left[\left(180\frac{\partial t}{\partial \Delta}V_P\right)/(\pi R)\right]^2}}{V_P},\quad (\Delta\geqslant 1.5^\circ)$$

或

$$\delta_{t_h}=\frac{z_S/z_e}{V_P\sqrt{(111.195\Delta)^2+z_e^2}},\quad (\Delta<1.5^\circ).$$

(5-3.1)

式中 $Z_S$台站海拔高程，以公里计。

$Z_e$震源深度，以公里计。

$V_P$ 表层纵波速度，可取为5.5km/s。

$R$ 地球平均半径，6371.0km。

$\Delta$ 震中距，以度为单位。

$\frac{\partial t}{\partial \Delta}$走时曲线斜率(又称慢度)，单位为秒/度，它可以通过走时曲线用数值微分法求得。

走时的地球椭圆扁率校正公式如下

$$\delta t_E = f(\Delta)(h_e + h_s). \qquad (5\text{-}3.2)$$

表5-3-1 高度差($h_s$或$h_e$)函数表

| φ | $h$(km) | φ | $h$(km) |
|---|---|---|---|
| 0°00′ | +7 | 44°55′ | −4 |
| 10°11′ | +6 | 47°37′ | −5 |
| 16°12′ | +5 | 50°17′ | −6 |
| 20°42′ | +4 | 53°01′ | −7 |
| 24°29′ | +3 | 55°50′ | −8 |
| 27°53′ | +2 | 58°46′ | −9 |
| 31°00′ | +1 | 61°51′ | −10 |
| 33°58′ | 0 | 65°11′ | −11 |
| 36°48′ | −1 | 68°52′ | −12 |
| 39°33′ | −2 | 73°11′ | −13 |
| 42°15′ | −3 | 78°50′ | −14 |
| 44°55′ | | 90°00′ | |

式中$h_e$为震中处平均椭球与等体积圆球之间的高度差，它是地理纬度的函数(见表5-3-1)。

$h_s$台站处按同上定义的高度差，也是地理纬度的函数。上述两值可分别由震中或台站的地理纬度查得。

$f(\Delta)$是与震中距有关的函数，可由表5-3-2查出。

表5-3-2 $f(\Delta)$函数表

| $\Delta$ | $f(\Delta)$* |
|---|---|
| 9° | 0.01 |
| 13° | 0.02 |
| 22° | 0.03 |
| 41° | 0.04 |
| 63° | 0.05 |
| 72° | 0.06 |
| 83° | 0.07 |
| 124° | 0.08 |
| 140° | *单位：s |

## (二) 震中距Δ的计算

在作震源参数修定时，要计算震中到各台站的距离$\Delta_i$，再

由$\Delta_i$从走时表查得理论走时，与实际观测走时相比较。根据残差以确定各震源参数的校正值。

设地震台的地理纬度为$\phi_s$，经度为$\lambda_s$，地心余纬度为$\phi_s'$；地震震中的地理纬度为$\phi_e$，经度为$\lambda_e$，地心余纬度为$\phi_e'$。则可按下式计算震中距$\Delta$：

$$\cos\Delta = aA + bB + cC, \quad (0^\circ \leqslant \Delta \leqslant 180^\circ). \qquad (5\text{-}3.3)$$

式中

$$a = \sin\phi_e'\cos\lambda_e,$$

$$b = \sin\phi_e'\sin\lambda_e,$$

$$c = \cos\phi_e',$$

$$A = \sin\phi_s'\cos\lambda s,$$

$$B = \sin\phi_s'\sin\lambda s,$$

$$C = \cos\phi_s'.$$

地心纬度$\overline{\phi}$与地理纬度$\phi$的关系为

$$\tan\overline{\phi} = (1-e)^2\tan\phi. \qquad (5\text{-}3.4)$$

式中　$e = 1/298.25$为地球扁率。

地心余纬度的定义如下：

$$\phi' = \begin{cases} 90^\circ - \overline{\phi}, & (\text{北纬}), \\ -90^\circ + \overline{\phi}, & (\text{南纬}). \end{cases} \qquad (5\text{-}3.5)$$

## （三）震源参数的修定

由(5-3.3)式计算的第$i$台震中距$\Delta_i$，可以按照对应于震源深度的走时表用内插方法得到理论计算走时$t_c$。于是第$i$台的走时残差$\xi_i$便是观测走时与理论走时之差

$$\xi_i = (t_{P\,i} - t_0) - t_{C\,i},$$

$$i = 1, 2, \cdots\cdots, n. \qquad (5\text{-}3.6)$$

式中$t_{Pi}$为第$i$台的$P$波到时，$t_0$为发震时刻。显然，残差出现是由于震中经纬度、震源深度与发震时刻的误差所引起，即可以将残差与各种误差之间的关系表示为

$$\xi_i=\frac{\partial t_{Ci}}{\partial t_0}\delta t_0+\frac{\partial t_{Ci}}{\partial \lambda_e}\partial\lambda_e+\frac{\partial t_{Ci}}{\partial \phi_e}\delta\phi_e+\frac{\partial t_{Ci}}{\partial z_e}\delta z_e,$$

$$i=1,2,\cdots,\ n. \qquad (5\text{-}3.7)$$

上式中各偏导数项分别为

$$\frac{\partial t_{Ci}}{\partial t_0}=1,$$

$$\frac{\partial t_{Ci}}{\partial \lambda_C}=\frac{\partial t_{ei}}{\partial \Delta_i}\ \frac{\partial \Delta_i}{\partial \lambda_e},$$

$$\frac{\partial t_{Ci}}{\partial \phi_e}=\frac{\partial t_{ei}}{\partial \Delta_i}\ \frac{\partial \Delta_i}{\partial \phi_e}. \qquad (5\text{-}3.8)$$

且有以下关系

$$\frac{\partial \Delta_i}{\partial \lambda_e}=-\sin z_i\ \sin\phi_e',$$

$$\frac{\partial \Delta_i}{\partial \phi_e}=\cos z_i. \qquad (5\text{-}3.9)$$

式中

$$\sin z_i=(\sin\phi_s'(\cos\lambda_e\sin\lambda_s-\sin\lambda_e\cos\lambda_s))/\sin\Delta_i,$$

$$\cos z_i=(\cos\phi_s'-\cos\phi_e'\cos\Delta_i)/(\sin\phi_e'\sin\Delta_i). \qquad (5\text{-}3.10)$$

现在解释式(5-3.7)—(5-3.10)中各符号的意义：

$z_i$　震中相对于第$i$台的方位角 $0°\leqslant z\leqslant 360°$。

$\frac{\partial t_{Ci}}{\partial \Delta_i}$　走时曲线的斜率，即慢度(单位为秒/度)，可以走时表上由内插公式计算。

$\frac{\partial t_{Ci}}{\partial z_e}$　走时对震源深度的微商，也可由走时表用内插公式求

得。

将(5-3.8)及(5-3.9)式中各有关项代入(5-3.7)式，可以得到下述修定震源参数的超定方程组

$$\delta t_0 + \left(\frac{\partial t_{C_i}}{\partial \Delta_i}\right)\cos z_i \delta\phi_e - \frac{\partial t_{C_i}}{\partial \Delta_i}\sin z_i \sin\phi'_e \delta\lambda_e + \frac{\partial t_{C_i}}{\partial z_e}\delta z_e = \xi_i,$$

$$i = 1,\ 2,\ \cdots,\ n. \tag{5-3.11}$$

如将上述方程组改写为矩阵形式则为

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}. \tag{5-3.12}$$

其中

$$\boldsymbol{A} = \begin{pmatrix} 1 & \frac{\partial t_{C_1}}{\partial \Delta_1}\cos z_1 & -\frac{\partial t_{C_1}}{\partial \Delta_1}\sin z_1 \sin\phi'_e & \frac{\partial t_{C_1}}{\partial z_e} \\ 1 & \frac{\partial t_{C_2}}{\partial \Delta_2}\cos z_2 & -\frac{\partial t_{C_2}}{\partial \Delta_2}\sin z_2 \sin\phi'_e & \frac{\partial t_{C_2}}{\partial z_e} \\ & \cdots & \cdots & \cdots \\ 1 & \frac{\partial t_{C_n}}{\partial \Delta_n}\cos z_n & -\frac{\partial t_{C_n}}{\partial \Delta_n}\sin z_n \sin\phi'_e & \frac{\partial t_{C_n}}{\partial z_e} \end{pmatrix}_{n\times 4}, \tag{5-3.13}$$

$$\boldsymbol{x} = (\delta t_0, \delta\phi_e,\ \delta\lambda_e,\ \delta z_e)^T, \tag{5-3.14}$$

$$\boldsymbol{b} = (\xi_1,\ \xi_2,\ \cdots,\ \xi_n)^T. \tag{5-3.15}$$

显然，系数矩阵$\boldsymbol{A}$及右端项$\boldsymbol{b}$均可根据震源参数的初值由走时表算出。根据广义逆方法，可得出震源参数修正值$\boldsymbol{x}$的解为

$$\boldsymbol{x} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}. \tag{5-3.16}$$

由此，得到修正后的震源参数为

$$\begin{aligned} t'_0 &= t_0 + \delta t_0, \\ \phi'_e &= \phi_0 + \delta\phi_e, \\ \lambda'_e &= \lambda_e + \delta\lambda_e, \\ z'_e &= z_e + \delta z_e. \end{aligned} \tag{5-3.17}$$

整个计算过程以下列方式迭代进行，每一次都用修定后的参

数，作为下一次计算的初值，反复计算。直到相邻两次计算结果的走时残差的标准误差(以秒为单位)的差值小于某一限差为止。残差的标准误差定义为

$$\mu = \pm\sqrt{\sum_{i=1}^{n} \xi_i^2/(n-m)}\ . \qquad (5\text{-}3.18)$$

$n$ 为所用的地震台数，$m$为修定的参数值，一般$m=4$。若不修定震源深度 $z_e$，$m=3$。

而修定值$t_0'$，$\phi_e'$，$\lambda_c'$及$z_e'$的均方误差为

$$\mu_{t0} = \pm\sqrt{|a^{-1}_{11}|},$$

$$\mu_{\phi} = \pm\mu\sqrt{|a^{-1}_{22}|},$$

$$\mu_{\lambda} = \pm\mu\sqrt{|a^{-1}_{33}|},$$

$$\mu_{ze} = \pm\mu\sqrt{|a^{-1}_{44}|}. \qquad (5\text{-}3.19)$$

式中 $a_{ii}^{-1}$ 为逆阵$(\boldsymbol{A}^T\boldsymbol{A})^{-1}$的主对角元素。

## §5-4 阻尼最小二乘法修定震源参数

由于震源参数的目标函数具有残差平方和形式，故用求目标函数的一阶导数的方法进行修定是合适的。以下将以近震修定为例进行讨论。

设第 $i$ 台$P$ 波到时的观测值为 $T_i$，由给定的震源参数$\boldsymbol{x}$求出之理论到时为$\tau_i(\boldsymbol{x})$，故残差

$$R_i(\boldsymbol{x}) = T_i - \tau_i(\boldsymbol{x}),\ i=1,\ 2,\ \cdots,\ n. \qquad (5\text{-}4.1)$$

而目标函数为

$$\Phi(\boldsymbol{x}) = \sum_{i=1}^{n} R_i^2(\boldsymbol{x}). \qquad (5\text{-}4.2)$$

式中$\boldsymbol{x}=(x, y, z, T)^T$，若对于远震则$\boldsymbol{x}=(\lambda, \phi, z, T)^T$。从某一初始点$\boldsymbol{x}_0$出发，用迭代法依次求各次近似值$\boldsymbol{x}_1$，$\boldsymbol{x}_2$，…，$\boldsymbol{x}_k$，…。其相应的校正向量$\Delta\boldsymbol{x}=(\Delta x, \Delta y, \Delta z, \Delta T)^T$可由(2-6.21)式计算

$$\Delta\boldsymbol{x}=(\boldsymbol{A}^T\boldsymbol{A}+\nu^2\boldsymbol{I})^{-1}\boldsymbol{g}.$$

式中$\nu^2$为阻尼因子，$\boldsymbol{I}$为单位矩阵，$\boldsymbol{A}$为如下形式之微分系数矩阵：

$$\boldsymbol{A}=\begin{bmatrix} \frac{\partial\tau_1}{\partial x} & \frac{\partial\tau_1}{\partial y} & \frac{\partial\tau_1}{\partial z} & \frac{\partial\tau_1}{\partial T} \\ \frac{\partial\tau_2}{\partial x} & \frac{\partial\tau_2}{\partial y} & \frac{\partial\tau_2}{\partial z} & \frac{\partial\tau_2}{\partial T} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial\tau_n}{\partial x} & \frac{\partial\tau_n}{\partial y} & \frac{\partial\tau_n}{\partial z} & \frac{\partial\tau_n}{\partial T} \end{bmatrix}_{\boldsymbol{x}=\boldsymbol{x}_0}, \tag{5-4.3}$$

而向量$\boldsymbol{g}$为：

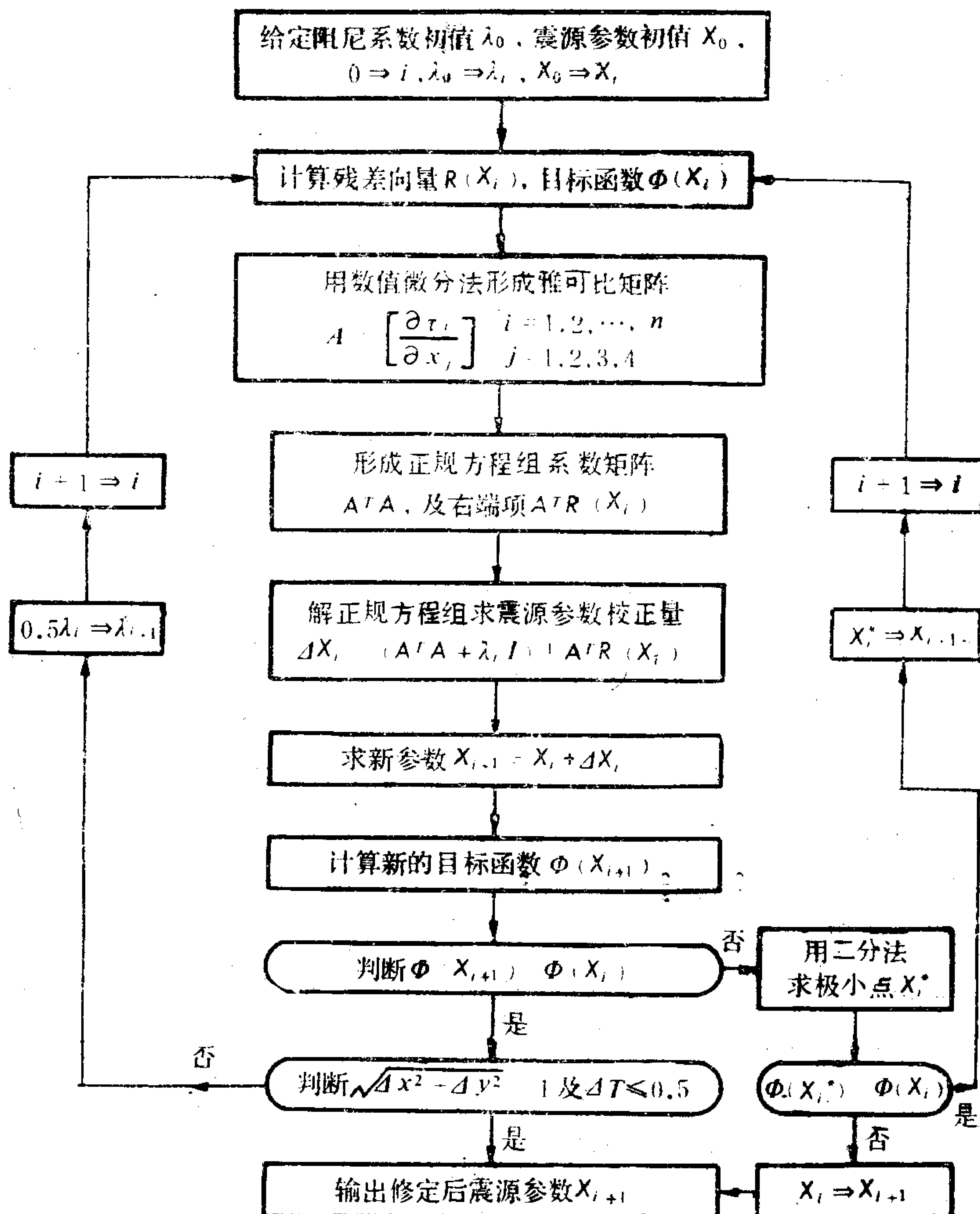
$$\boldsymbol{g}=\boldsymbol{A}^T[\boldsymbol{T}-\tau(\boldsymbol{x})]. \tag{5-4.4}$$

其中$\boldsymbol{T}=(T_1, T_2, \cdots, T_n)^T$，$\tau(\boldsymbol{x})=(\tau_1(\boldsymbol{x}), \tau_2(\boldsymbol{x}), \cdots \tau_n(\boldsymbol{x}))^T$，分别为到时的观测值及理论值向量。

微分系数矩阵$\boldsymbol{A}$的计算可利用走时表进行内插，即对四个待定的震源参数$x$，$y$，$z$，$T$分别增大$10^{-2}$或$10^{-3}$，考察相应走时的变化，并用(2-6.33)式进行计算。

按照式(5-4.1)—(5-4.4)迭代计算，直到目标函数$\Phi(\boldsymbol{x})$达到预定精度为止。除了用走时残差平方和作为评定准则外，还可以采用校正值的大小作为收敛条件。例如，对于近震，当震中距校正量$\sqrt{\Delta x^2+\Delta y^2}\leqslant 1\text{km}$，发震时刻校正量$\Delta T\leqslant 0.5\text{s}$；远震震中距校正量$\sqrt{\Delta\phi^2+\Delta\lambda^2}\leqslant 0.1°$，发震时刻校正量$\Delta T\leqslant 1\text{s}$时，则停止迭代。

用阻尼最小二乘法修定基本震源参数的计算框图如图5-4-1示。本方法对远震震中修定也同样适用。



给定阻尼系数初值 $\lambda_0$，震源参数初值 $X_0$，
$0 \Rightarrow i$，$\lambda_0 \Rightarrow \lambda_i$，$X_0 \Rightarrow X_i$
计算残差向量 $R(X_i)$，目标函数 $\Phi(X_i)$
用数值微分法形成雅可比矩阵
$A = \left[\frac{\partial \tau_i}{\partial x_j}\right]$ $i=1,2,\cdots,n$ $j=1,2,3,4$
形成正规方程组系数矩阵
$A^TA$，及右端项 $A^TR(X_i)$
解正规方程组求震源参数校正量
$\Delta X_i = (A^TA+\lambda_i I)^{-1}A^TR(X_i)$
求新参数 $X_{i+1}=X_i+\Delta X_i$
计算新的目标函数 $\Phi(X_{i+1})$
判断 $\Phi(X_{i+1}) < \Phi(X_i)$
否
用二分法
求极小点 $X_i^*$
是
判断 $\sqrt{\Delta x^2+\Delta y^2} \leqslant 1$ 及 $\Delta T \leqslant 0.5$
否
$0.5\lambda_i \Rightarrow \lambda_{i+1}$
$i+1 \Rightarrow i$
$\Phi(X_i^*) < \Phi(X_i)$
是
否
$X_i^* \Rightarrow X_{i+1}$
$i+1 \Rightarrow i$
是
输出修定后震源参数 $X_{i+1}$
$X_i \Rightarrow X_{i+1}$


图5-4-1

## § 5-5 Powell法修定震源参数

Powell 方法是一种直接寻查目标函数极小值的有效方法。该方法的原理及具体算法已在第二章 § 2-5 中详细讨论过。这里仅讨论具体应用于修定震源参数时的一些问题。

同样地，设第$i$台$P$波到时的观测值为$T_i$，由给定的（初值或迭代值）震源参数$\boldsymbol{x}$求出之理论到时为$\tau_i(\boldsymbol{x})$，由此定义的目标函数为二次型函数

$$\Phi(\boldsymbol{x})=\sum_{i=1}^{n}[T_i-\tau_i(\boldsymbol{x})]^2. \qquad (5\text{-}5.1)$$

式中对近震$\boldsymbol{x}=(x, y, z, T)^T$,对远震$\boldsymbol{x}=(x, \phi, z, T)^T$。

我们知道，对于二次型的极值问题，共扼梯度法是一个有效的方法。作为共扼梯度法的改进的Powell法，对于求(5-5.1)式的目标函数$\Phi(\boldsymbol{x})$的极小值是很有用的。这时，只须将 § 2-5 中所述的步骤及框图进行计算即可。

在作一维寻查时，其步长$STEP$的选取与具体问题中自变量单位有关。对于近震，其单位为公里，远震单位为度(或弧度)。一般而言，若$\boldsymbol{p}=(p_1, p_2, \cdots, p_n)^T$是一个寻查方向。则可以取$STEP=E(p_1^2+p_2^2+\cdots+p_n^2)^{-1/2}$作为步长。例如，近震可取$E=1$公里，远震可取$E=0.1$度等。

结束迭代的收敛准则可以采用迭代前后两次的目标函数的差值达到预定的限差

$$|\Phi(\boldsymbol{x}^*)-\Phi(\boldsymbol{x}_0)|\leqslant\varepsilon_\phi,$$

或迭代前后两点的欧式距离

$$\|\boldsymbol{x}^*-\boldsymbol{x}_0\|\leqslant\varepsilon_x$$

等条件。一般取$\varepsilon_\phi=10^{-4}—10^{-6}$,（单位：秒$^2$）$\varepsilon_x=10^{-3}—10^{-5}$（单位：度或公里）。

## §5-6 试探法求近震震源深度

在实际修定震源参数时，由于震源深度与其它几个参数是不独立的，震中位置或发震时刻的少量变化，可引起震源深度的较大误差。所以震源深度是较难求准的。为了提高震源深度的计算精度，采用了分别修定的办法。一般步骤是，先用上述的Geiger法或阻尼最小二乘法以及Powell法等，对$x$，$y$，$T$（或$\lambda$，$\varphi$，$T$）三个参数进行修定。在计算过程中一般是将震源的某一深度值$z$固定。在求得较精确的震中及发震时刻后，再利用下述试探法求震源深度。

从$n$个实际记录到$P$波到时的地震台中挑选出三个距震中最近的台，分别求这三台的震中距$D_i(i=1，2,3)$，然后假设一系列震源深度（通常取0，1，2，3，…,50公里），计算相应的震源距

$$S_i^{设}(z_j)=\sqrt{D_i^2+z_j^2},$$

$$i=1，2，3，j=0，1，2，3，\cdots，50，$$

根据这三个台的走时$T_i-T$，利用走时表可以计算出各相应深度的震源距理论值$S_i^{理}$。将假设的震源距$S_i^{设}(z_i)$与$S_i^{理}$比较，计算这三个台的震源距残差平方和

$$\varepsilon_j=\sum_{i=1}^{3}\left[S_i^{设}(z_j)-S_i^{理}\right]^2,$$

$$j=0，1，2，3，\cdots，50.$$

对这一系列残差平方和$\varepsilon_j$值进行比较，取$\varepsilon_j$最小值所对应的深度$z$工作为最后修定的震源深度。

## § 5-7 非均匀地壳模型中地震波走时的计算

前面我们仅考虑了简单地壳模型中的震源定位问题。即假定地壳是单层或双层的均匀模型。这时，计算从震源到台站的走时问题特别简单，可以用解析公式或走时表即可求得。

然而实际地壳结构是非均匀的，地下介质的速度无论沿纵向或横向都有较复杂的变化。在这种情况下，地震射线呈现为复杂的折线或曲线形状，走时不能简单地从解析公式或走时表获得。一般说来，只有用射线追踪方法，才能获得任意两点间地震波的走时，作精确的定位计算。下面讨论非均匀介质中地震波走时计算原理。

设任意介质中速度$v$是空间坐标的连续函数$v(x, y, z)$，并且有连续一阶偏微分。根据 Fermat 原理，地震射线的方程可以表达为：

$$
\begin{aligned}
&\frac{d}{ds}\left(\frac{1}{v}\ \frac{dx}{ds}\right)-\frac{\partial}{\partial x}\left(\frac{1}{v}\right)=0,\\
&\frac{d}{ds}\left(\frac{1}{v}\ \frac{dy}{ds}\right)-\frac{\partial}{\partial y}\left(\frac{1}{v}\right)=0, \qquad (5\text{-}7.1)\\
&\frac{d}{ds}\left(\frac{1}{v}\ \frac{dz}{ds}\right)-\frac{\partial}{\partial z}\left(\frac{1}{v}\right)=0.
\end{aligned}
$$

通常有两种方法解非均匀介质中射线追踪问题。一种是解一系列从一个端点出发的初值问题，并设计一种方案以使射线会聚到另一端点。另一种途径是直接解任意两点的边值问题。后一种方法在计算上较快。下面只讨论边值问题的解法。

两点边值问题实质上是寻求满足两个端点的射线方程（5-7.1)式的解答。一种应用较广的方法是将上述二阶方程组化为一阶方程组。然后用有限差分程序来求解。其方法如下：

引入位置矢量$r$，及慢度$u(r)$

$$r = \begin{pmatrix} x \\ y \\ z \end{pmatrix}, \tag{5-7.2}$$

$$u(r) = \frac{1}{v(r)}. \tag{5-7.3}$$

（慢度 $u(r)$ 及速度 $v(r)$ 都是空间函数）。

于是射线方程（5-7.1）给出的一般式可写为

$$\frac{d}{ds}\left(u\frac{dr}{ds}\right) = \nabla u. \tag{5-7.4}$$

其中 $\nabla u$ 表示$u$ 的梯度，

$$\nabla u = \begin{pmatrix} \partial u/\partial x \\ \partial u/\partial y \\ \partial u/\partial z \end{pmatrix}. \tag{5-7.5}$$

（5-7.4）式又可写为

$$\frac{d^2 r}{ds^2} = \frac{1}{u}\left(\nabla u - \frac{dr}{ds}\frac{du}{ds}\right). \tag{5-7.6}$$

以 $G$表示慢度对射线路径$S$ 的微分，则可写为以下表达式

$$G = \frac{du(r)}{ds} = \frac{\partial u}{\partial x}\frac{dx}{ds} + \frac{\partial u}{\partial y}\frac{dy}{ds} + \frac{\partial u}{\partial z}\frac{dz}{ds}$$

$$= u_x\dot{x} + u_y\dot{y} + u_z\dot{z}. \tag{5-7.7}$$

其中，$\dot{x}$，$\dot{y}$，$\dot{z}$ 分别表示 $x$，$y$，$z$ 对 $s$ 的微分，即方向余弦；$u_x = \partial u/\partial x$，$u_y = \partial u/\partial y$，$u_z = \partial u/\partial z$，为慢度变化率。因此，方程(5-7.6)可以分别写为以下三个二阶微分方程：

$$\begin{aligned} \ddot{x} &= v(u_x - G\dot{x}), \\ \ddot{y} &= v(u_y - G\dot{y}), \\ \ddot{z} &= v(u_z - G\dot{z}). \end{aligned} \tag{5-7.8}$$

若我们用矢量$w$表示要求解的变量，其各分量分别代表。$w_1 = x$，$w_2 = \dot{x} = \frac{\partial x}{\partial s}$，$w_3 = y$，$w_4 = \dot{y} = \frac{\partial y}{\partial s}$，$w_5 = z$，$w_6 = \dot{z} = \frac{\partial z}{\partial s}$。则(5-7.8)式与下面六个一阶微分方程等价：

$$
\begin{aligned}
\dot{w}_1 &= w_2, \\
\dot{w}_2 &= v(u_x - G w_2), \\
\dot{w}_3 &= w_4, \\
\dot{w}_4 &= v(u_y - G w_4), \\
\dot{w}_5 &= w_6, \\
\dot{w}_6 &= v(u_z - G w_6).
\end{aligned}
\tag{5-7.9}
$$

这里$G$由(5-7.7)式给出

$$G = u_x w_2 + u_y w_4 + u_z w_6. \tag{5-7.10}$$

并注意到方向余弦平方和有关系：

$$\left(\frac{\partial x}{\partial s}\right)^2 + \left(\frac{\partial y}{\partial s}\right)^2 + \left(\frac{\partial z}{\partial s}\right)^2 = w_2^2 + w_4^2 + w_6^2 = 1. \tag{5-7.11}$$

由于我们需要计算的是从$A$点到$B$点的走时$T = \int_A^B u ds$。为此，引入一个附加的变量$w_7$，以代表沿射线路径某一点的走时$\tau$。相应有关系$\dot{w}_7 = u$。设$s'$为从$A$点到$B$点整个路径的长度，则总的走时$T = \tau_{B点} = w_7(s')$。再引入一个变量$w_8 = s'$，与其相应的微分方程为$\dot{w}_8 = 0$。为了计算方便，把弧长$s$归一化，使其取值在0和1之间，我们引入一个新的变量$t$来代替$s$

$$t = s/s'. \tag{5-7.12}$$

这样，各变量分别对$t$求微分便得到下面八个一阶微分方程：

$$w_1' = w_8 w_2,$$

$$
\begin{aligned}
w_2' &= w_8 v(u_x - G w_2), \\
w_3' &= w_8 w_4, \\
w_4' &= w_8 v(u_y - G w_4), \\
w_5' &= w_8 w_6, \\
w_6' &= w_8 v(u_z - G w_6), \\
w_7' &= w_8 u, \\
w_8' &= 0.
\end{aligned}
\tag{5-7.13}
$$

由这些方程得到的解相应的变量为

$$
\begin{aligned}
w_1 &= x, & w_2 &= dx/ds, \\
w_3 &= y, & w_4 &= dy/ds, \\
w_5 &= z, & w_6 &= dz/ds, \\
w_7 &= \tau, & w_8 &= s'.
\end{aligned}
\tag{5-7.14}
$$

微分方程组(5-7.13)的边界条件为

$$
\begin{aligned}
&w_1(0) = x_A,\ w_3(0) = y_A,\ w_5(0) = z_A, \\
&w_1(1) = x_B,\ w_3(1) = y_B,\ w_5(1) = z_B, \\
&w_7(0) = 0,\ w_2^2(0) + w_4^2(0) + w_6^2(0) = 1.
\end{aligned}
\tag{5-7.15}
$$

式中 $(x_A,\ y_A,\ z_A)$为$A$点坐标,$(x_B,\ y_B,\ z_B)$为$B$点坐标。第七个边界条件表示初始点$A$的局部走时为零，第八个边界条件表示射线方向余弦的平方和为1，即式(5-7.11)。

求解一阶微分方程(5-7.13)，一般利用有限差分法计算。

在地震定位问题中，对于给定的地壳模型，需要计算震源到接收站之间的最小走时。实际上,最小走时$\tau$正是从一阶微分方程组(5-7.13)解出的变量之一。然而,用这种方法计算走时要付出高昂的计算费用。因此,更常用的方法是采用简化的速度模型。例如，假定速度随深度加大而增高的情况，$v = v(z)$。这时，射线路径变为一个圆弧，可以很方便追索射线路径及计算走时$T$，其

计算方法可参见第十章。另一种是假定地下为一系列水平层，每一层速度恒定或具有一定垂直梯度值，计算位于任何深度的震源到地面台站的走时，可以根据解析公式或用迭代法计算，具体算法可参见第六章。

## §5-8 近震定位计算程序

下面给出一个用FORTRAN-77语言编写的近震定位的计算机程序。根据使用者的需要，将全国或者区域台网的地震台站的经纬度、高程事先输入计算机内。计算时将有关地震台号及*P*波到时依次输入，利用区域近震走时表进行修定，最后输出修定的震源参数。计算流程见图5-4-1。

### （一）标识符及子程序说明

N——记录某一地震的台站数。

NJC——选择某一震源深度相应走时表，NJC为1，2，3分别相应为0，10，16公里。

DATE——发震日期。

MAG——地震震级。

X，Y，Z——直角坐标系震中位置及震源深度。

T——发震时刻。

TRO，DS——工作单元。

LONGIO——直角坐标系原点的经度λ。

LATIO——直角坐标系原点的纬度φ。

LONGI——震中经度λ。

LATI——震中纬度φ。

RN，RM——变换坐标时的地球参数。

PIMN——定震源参数时的走时标准误差。

STATI(21, 3)　北京地区地震台网21个台站的经纬度及高程。

DEPTH(3)　近震走时表的三个震源深度，分别为0,10,16公里。

DELTA(104)　近震走时表的震中距。

BJC(3, 104)　三种不同震源深度的近震走时表（以秒为单位）。

FJC(104)　表示某一震源深度的走时表，作中间工作单元用。

STAT2　记录到某次地震的日期，震级及N个台站的编号及初至波到时。

STATD　日期、震级、台站编号及波的到时的数组。

XB, YB, ZB　记录到地震的N个台站的直角坐标$x_i$，$y_i$ $z_i$数组。

TP　$N$个台站的初至波到时。

TAO　$N$个台站计算的理论到时。

BJ　形成初值计算超定方程组的增广系数矩阵。

DJ　正规方程组增广系数矩阵。

XYZT　存放初始及修定后震源参数$x$，$y$，$z$，$T$的数组。

RESID　各台走时的残差。

BJAC　用数值微分法形成的 Jacobi 矩阵。

STATN　台站编号数组。

STANA　台站名数组。

ISS　是否输出原始数据的控制标量数组。

SECOND(HOUR)　将以时分秒为单位的到时 HOUR 化为以秒为单位。

HOUR(SECOND)　将秒SECOND化为时分秒。

RADIAN(A, B, C)　将度(A)，分(B)，秒(C)化为弧度。

DEGREE(A1, B, C, D)——将弧度(D)化为度(A1), 分(B), 秒(C)。

GAUS(A, N, N1)——用GAUSS法解线性方组组; A为增广系数矩阵, N为方程组阶数, N1为右端项数。

MOND(AM, CM, M, N)——将超定方程组化为正规方程组; AM为超定方程组的增广系数矩阵, CM为正规方程组的增广系数矩阵, $M$为超定方程个数, $N$为未知数个数。

QIP(N, T, X, Y, F)——一元三点插值过程; N为结点数, T为内插的自变量, F为内插的函数值, X、Y为结点的自变量及函数的数组。

PREVAL——初定震源参数X、Y、Z、T的过程。

LFCHXY——将$\lambda$, $\phi$化为$X$, Y坐标。

CHECKS(KK)——检查台站数据的过程; KK为剔除的台数。

OUTLFH——将震源参数$X$、$Y$、$Z$、$T$化为$\lambda$, $\varphi$, $Z$, $T$并输出的过程。

EXPLOZ——用试探法计算震源深度$z$的过程。

FUNCT(U, F)——求各台站理论到时的过程; U为震源参数数组, F为理论到时数组。

CALPHI(PHI, F, DF)——计算走时残差及目标函数(残差平方和)的过程; $F$为各台站理论到时, PHI为目标函数$\Phi=\sum_{i=1}^{n} R_i^2$, DF为残差项$R_i$的数组。

MAQT(LMDO, UO, M)——用阻尼最小二乘法修定震源参数的过程;

LMDO——为阻尼因子初值;

UO——在计算开始时为震源参数初值, 计算结束时为震源参数的修定值;

M——为待修定参数的数目。

## （二）程序的使用

1．输入台站的经纬度及高程数组STATI，走时表三个不同震源深度数组 DEPTH，近震走时表的震中距 DELTA 以及走时BJC。

2．输入记录到某次地震用的台数 N，选择走时表的深度参数NJC。

3．输入发震日期，震级及各台编号、初至波到时的数组STATD。

## （三）FORTRAN-77语言程序及计算实例

```
C       THIS PROGRAM IS DESIGNED TO DETERM -
           INE THE FOCUS
C       PARAMETERS OF LOCAL EARTHQUAKE BY
C       USING THE FIRST ARRIVAL DATA ON THE
C       STATIONS
       IMPLICIT DOUBLE FRECISION(A-H,L,P-Z)
        INTEGER STATN
        REAL LONGIO,LATIO,TONGI,LATI,MAG
        COMMON /NN/N,NJC
        COMMON /XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
          RM,PIMN
        COMMON/SLM/MAG,LONGIO,LATIO,LONGI,LATI
        COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),TAO
          (30),
        STATD(31,2),BJ(30,4),DJ(3,4),XYZT(4),RESID
          (30),
        BJAC(30,5),STATN(30)
        COMMON/STAT/STATI(21,3),DEPTH(3),
        DELTA(104),BJC(3,104),FJC(104)
        COMMON/STA/STANA(21)
```

```
      CHARACTER*12 STANA
      DIMENSION ISS(15)
      DIMENSION STA3(63),BJC3(312),STAT2(60)
C
C     INPUTING THE NAME AND THE PARAMETERS
C     OF THE
C     STATION AND THE TRAVEL TIME TABLE
C
      OPEN(11,FILE='SWITCH' STATUS='OLD')
      READ(11,*)(ISS(I),I=1,15)
      WRITE(*,3000)(ISS(I),I=1,15)
3000  FORMAT(/27X,'***** SWITCH LIST *****'
         /18X,3(5I2,5X)/)
      OPEN(1,FILE='DAT1' )
      READ(1,*)(STANA(I),I=1,21)
      IF(ISS(1).EQ.0) GO TO 3001
      WRITE(*,2000)
2000  FORMAT(/25X,'***** STATION NAME ***
      **' /)
      WRITE(*,1000)(I,STANA(I),I=1,21)
1000  FORMAT(3(7X,I3,5X,A12))
3001  CONTINUE
      OPEN(2,FILE='DATAST')
      READ(2,*)(STA3(I),I=1,63)
      IF(ISS(2).EQ.0) GO TO 3002
1001  FORMAT(/20X,'PHI',12X,'LAMBAD',12X,'HIGH')
      WRITE(*,1001)
      DO 110 I=1,21
       I3=I*3
       I2=I3-1
       I1=I3-2
      WRITE(*, 120) STA3(I1), STA3(I2), STA3(I3
120   FORMAT(10X,3F16,6)
```

```
  110  CONTINUE
 3002  CONTINUE
       DO 20 I=1,21
       DO 15 J=1,3
   15  STATI(I,J)=STA3(3*(I-1)+J)
   20  CONTINUE
       OPEN(3,FILE='DATADP')
       READ(3,*)(DEPTH(I),I=1,3)
       IF(ISS(3).EQ.0) GO TO 3003
       WRITE(*,1002)
 1002  FORMAT(/37X,'~~DEPTH~~')
       WRITE(*,140)(DEPTH(I),I=1,3)
  140  FORMAT(14X,3F15,2)
 3003  CONTINUE
       OPEN(4,FILE='DATADS')
       READ(4,*)(DELTA(I),I=1,104)
       IF(ISS(4).EQ.0) GO TO 3004
       WRITE(*,1004)
 1004  FORMAT(/37X,'~~DELT'~~')
       WRITE(*,150)(DELTA(I),I=1,104)
  150  FORMAT(5F15.4)
 3004  CONTINUE
       OPEN(8,FILE='DATAJC')
       READ(8,*)(BJC3(I),I=1,312)
       IF(ISS(8).EQ.0) GO TO 3008
       WRITE(*,1005)
 1005  FORMAT(/36X,'~~TRAVEL TIME~~')
       DO 160 I=1,104
       I1=104+I
       I2=208+I
       WRITE(*,170) BJC3(I),BJC3(I1),BJC3(I2)
  170  FORMAT(14X,3F16,4)
```

```
 160  CONTINUE
3008  CONTINUE
      DO 50 I=1,3
      DO 40 J =1,104
  40  BJC(I,J)=BJC3(104 * (I-1)+J)
  50  CONTINUE
      DO 100 I=1, 21
      DO 102 J =1,2
      D=STATI(I,J)
      A=AINT(D)
      B=AINT((D-A) * 100. )
      C=((D-A) * 100.-B) * 100.
      STATI(I,J)=RADIAN(A,B,C)
 102  CONTINUE
 100  CONTINUE
      OPEN(5,FILE='NNJC')
      READ(5, * ) N,NJC
      IF(ISS(5).EQ.0) GOTO 3005
      WRITE( * ,1006)N,NJC
1006  FORMAT(21X,'N =',I3,15X,'NJC =',I3)
3005  CONTINUE
      DO 200 I =1,104
 200  FJC(I)=BJC(NJC,I)
      OPEN(10,FILE='DATARE')
      READ(10, * )(STAT2(I),I=1,2 * (N+1))
      IF(ISS(10).EQ.0) GO TO 3010
      WRITE( * ,1010)
1010  FORMAT(/20X,'STATION NUMBER',10X,'ARRIVAL
        TIME')
      DO 201 I=1,N+1
      I2=2 * I
      I1=I2-1
```

```
      WRITE( * ,202)STAT2(I1),STAT2(I2)
 202  FORMAT(14X,F15.1,10X,F15.3)
 201  CONTINUE
3010  CONTINUE
      DO 300 I=1,N+1
      DO 250 J=1,2
 250  STATD(I,J)=STAT2(2*(I-1)+J)
 300  CONTINUE
      DATE=STATD(1,1)
      MAG=STATD(1,2)
      DO 310 I=1,N
      TP(I)=STATD(I+1,2)
      STATN(I)=(STATD(I+1,1)+.01)
 310  CONTINUE
      DO 320 I=1,N
 320  TP(I)=SECOND(TP(I))
      TRO=TP(1)-1000.
      DO 330 I=1,N
 330  TP(I)=TP(I)-TRO
      CALL LFCHXY
      CALL PREVAL
      WRITE( * ,400)
 400  FORMAT(/,25X,'PRELIBMINARY VALUE')
      CALL OUTLFH
      CALL MAQT(1.,XYZT,4)
      X=XYZT(1)
      Y=XYZT(2)
      Z=XYZT(3)
      T=XYZT(4)
      CALL CHECKS(K)
      N=N-K
      IF(K.NE.0)THEN
```

```
      CALL MAQT(1.,XYZT,4)
      X=XYZT(1)
      Y=XYZT(2)
      Z=XYZT(3)
      T=XYZT(4)
      END IF
      CALL EXPLOZ
      WRITE(*,500)
500   FORMAT(///,25X,'MODIFIED VALUE')
      CALL OUTLFH
      DO 600 I=1,N
      TP(I)=TP(I)+TRO
600   TP(I)=HOUR(TP(I))
      WRITE(*,700)
700   FORMAT(/,16X,'CODE AND ARRIVAL TIME FOR',
       1X,'RECORDING STATIONS',/,12X,'STATION NA-
        ME',
        8X,'ARRIVAL TIME',5X,'RESIDUAL',/)
      DO 900 I=1,N
      M=STATN(I)
      WRITE(*,800)M,STANA(M),TP(I),RESID(I)
800   FORMAT(10X,I3,2X,1A12,5X,F10,1,8X,F4.1,)
900   CONTINUE
      STOP
      END
C
C
C
      FUNCTION RADIAN(A,B,C)
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      RADIAN(A+B/60.+C/3600.)*0.0174532925
      RETURN
```

```
      END
C
C
      FUNCTION SECOND(HOUR)
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      SECOND=AINT(HOUR*1E-4)*3600.+(AINT(HOU-
         R*1E-2)
         -(AINT(HOUR*1E-4)*100.))*60.+HOUR-(AINT
         (HOUR*1E-2)*100.)
      RETURN
      END
C
C
C
      FUNCTION HOUR(SECOND)
      IMPLICIT DOUBLE PRECISION(A-H,L,M,P-Z)
      REAL M
      TT=SECOND
      TT1=TT/3600.0
      H=AINT(TT1)
      TM=AINT((SECOND-H*3600.)/60.)
      S=SECOND-H*3600.-TM*60.
      HOUR=H*1E4+TM*1E2+S
      RETURN
      END
C
C
C
      SUBROUTINE DEGREE(A1,B,C,D)
      IMPLICIT DOUBLE PRECISION(A-I,P-Z)
      P=57.2957795
      D=D*P
      A1=AINT(D)
```

```
      A=A1
      D=(D-A)*60.0
      B=AINT(D)
      C=(D-B)*60.0
      A1=A
      RETURN
      END
C
C
C
      SUBROUTINE GAUS(A,N,N1)
      IMPLICIT DOUBLE PRECISION (A-G,P-Z)
      DIMENSION A(N,N+N1)
      INTEGER H
      H=1
   10 P=0.0
      DO 20 I=H,N
      DO 20 J=1,N
      IF(DABS(A(I,J)).GE.P)THEN
      P=DABS(A(I,J)
      K=I
      L=J
      END IF
   20 CONTINUE
      IF(P.LT.1E-30) THEN
      END IF
      DO 30 J=1,N+N1
      Q=A(K,J)
      A(K,J)=A(H,J)
      A(H,J)=Q
   30 CONTINUE
      DO 40 I=1,N
```

```
      Q = - A(I,L)/A(H,L)
      IF(I.NE.H)THEN
      DO 35 J =1,N+N1
   35 A(I,J) = A(I,J) + A(H,J) * Q
      END IF
   40 CONTINUE
      Q = A(H,L)
      DO 50 J =1,N+N1
   50 A(H,J) = A(H,J)/Q
      H = H + 1
      IF(H.LE,N) GO TO 10
      DO 60 J =1,N
      DO 60 I=1, N
      IF(A(I,J).GT.0.5) THEN
      DO 65 K=1,N+N1
      Q =A(I,K)
      A(I,K) =A(J,K)
   65 A(J,K) =Q
      END IF
   60 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE MOND(AM,CM,M,N)
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      DIMENSION AM(30,4),CM(N,N+1)
      DO 20 J =1,N
      DO 20 K=1,N+1
      IF(K.GE.J)THEN
      CM(J,K)=0.
```

```
      DO 10 I=1,M
   10 CM(J,K)=CM(J,K)+AM(I,J)*AM(I,K)
      END IF
   20 CONTINUE
      DO 30 J=1,N
      DO 30 K=1,N
      IF(J.LT.K)CM(K,J)=CM(J,K)
   30 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE QIP(N,T,X,Y,F)
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)
      DIMENSION X(N),Y(N)
      DO 10 I=1,N-3
      IF(T.LE.X(I+1)) GO TO 20
   10 CONTINUE
      I=N-2
      GO TO 30
   20 IF((I.NE.1).AND.(T-X(I).LT.X(I+1)-T)) I=I-1
   30 X0=X(I)
      X1=X(I+1)
      X2=X(I+2)
      U=(T-X1)*(T-X2)/((X0-X1)*(X0-X2))
      V=(T-X0)*(T-X2)/((X1-X0)*(X1-X2))
      W=(T-X0)*(T-X1)/((X2-X0)*(X2-X1))
      F=U*Y(I)+V*Y(I+1)+W*Y(I+2)
      RETURN
      END
C
C
```

```
C
      SUBROUTINE PREVAL
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      INTEGER STATN
      REAL LONGIO,LATIO,LONGI,LATI,MAG
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,RM,
      PIMN
      COMMON/SLM/MAG,LONGIO,LATIO, LONGI,LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),
         TAO(30),STATD(31,2),BJ(30,4),DJ(3,4),
         XYZT(4),RESID(30),BJAC(30,5),STATN(30)
      VP=6.
      PIMN=0.
      DS=XB(1)*XB(1)+YB(1)*YB(1)-VP*VP*TP(1)*
      TP(1)
      DO 10 I=1,N-1
      BJ(I,1)=XB(I+1)-XB(1)
      BJ(I,2)=YB(I+1)-YB(1)
      BJ(I,3)=-VP*VP*(TP(I+1)-TP(1))
      BJ(I,4)=(XB(I+1)**2+YB(I+1)**2-VP**2*TP(I+
      1)**2-DS)/2.
   10 CONTINUE
      CALL MOND(BJ,DJ,N-1,3)
      CALL GAUS(DJ,3,1)
      X=DJ(1,4)
      XYZT(1)=X
      Y=DJ(2,4)
      XYZT(2)=Y
      T=DJ(3,4)
      XYZT(4)=T
      Z=0.0
```

```
      DO 20 I=1,N
      DS=(VP*(TP(I)-T))**2-(XB(I)-X)**2-(YB(I)-
     Y)**2
      IF(DS.LE.0.)GOTO 20
      Z=Z+DSQRT(DS)
   20 CONTINUE
      Z=Z/FLOAT(N)
      XYZT(3)=Z
      RETURN
      END
C
C
C
      SUBROUTINE LFCHXY
      IMPLICIT DOUBLE PRECISION (A-H,L,P-R,S-Z)
      INTEGER STATN
      REAL LONGIO,LATIO,LONGI,LATI,MAG
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,RM,
      PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),
     TAD(30),STATD(31,2),BJ(30,4),DJ(3,4),
     XYZT(4),RESID(30),BJAC(30,5),STATN(30)
      COMMON/STAT/STATI(21,3),
     DEPTH(3),DELTA(104),BJC(3,104),FJC(104)
      COMMON/STA/STANA(21)
      CHARACTER*12 STANA
      BB1=0.0
      BB2=0.0
      LATIO=BB1
      LONGIO=BB2
```

```
      DO 10 I=1,N
      J=STATN(I)
      LONGIO=LONGIO+STATI(J,2)
      LATIO=LATIO+STATI(J,1)
   10 CONTINUE
      FN=FLOAT(N)
      LONGIO=LONGIO/FN
      LATIO=LATIO/FN
      A=6378.160
      EE=0.0066935176
      D=DSQRT(1.-EE*SIN(LATIO)**2)
      RN=A/D
      RM=A*(1.-EE)/D**3
      DO 20 I=1,N
      J=STATN(I)
      LONGI=STATI(J,2)
      LATI=STATI(J,1)
      XB(I)=RN*(LONGI-LONGIO)*COS(LATI)
      YB(I)=RM*(LATI-LATIO)+(XB(I)**2/(2*RN)
     *  *TAN(LATIO))
      ZB(I)=STATI(J,3)
   20 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE EXPLOZ
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      INTEGER STATN
      REAL LONGIO,LATIO,LONGI,LATI,MAG
      DIMENSION ERZ(0:50),TPZ(3),XBZ(3),YBZ(3),
```

```
      ZBZ(3)
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
         RM,PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,
         LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),TAO,
         (30),
         STATD(31,2),BJ(30,4),DJ(3,4),XYZT(4),RESID
         (30), BJAC(30,5),STATN(30)
      COMMON/STAT/STATI(21,3),DEPTH(3),
         DELTA(104),BJC(3,104),FJC(104)
      COMMON/STA/STANA(21)
      CHARACTER*12 STANA
      DO 100 J=1,3
      TPZ(J)=TP(J)
      DO 50 K=J+1,N
      IF(TP(K).LT.TPZ(J)) THEN
      TPZ(J)=TP(K)
      XBZ(J)=XB(K)
      YBZ(J)=YB(K)
      ZBZ(J)=ZB(K)
      END IF
 50   CONTINUE
100   CONTINUE
      DO 200 I=0,50
200   ERZ(I)=0
      DO 300 J=0,50
      DO 250 I=1,3
      Z=-FLOAT(J)
      TRA=TPZ(I)-T
      CALL QIP(104,TRA,FJC,DELTA,DS)
```

```
      ERZ(J)=ERZ(J)+((XBZ(I)-X)**2+(YBZ(I)-Y)
     **2
     1+(ZBZ(I)-Z)**2-(DS**2+DEPTH(NJC)**2))**2
  250 CONTINUE
  300 CONTINUE
      BMU=ERZ(0)
      Z=0.
      DO 400 J =0,50
      IF(ERZ(J).LT.BMU) THEN
      BMU=ERZ(J)
      Z=J
      END IF
  400 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE OUTLFH
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      INTEGER STATN
      REAL LONGIO,LATIO,LONGI,LATI,LA,LO,
        MAG
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
        RM,PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,
        LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),TAO
        (30),
        STATD(31,2),BJ(30,4),DJ(3,4),XYZT(4),RESID
        (30), BJAC(30,5),STATN(30)
```

```
      LATI=LATIO+Y/RM-(X*X*TAN(LATIO))/(2*
         RM*RN)
      S1=LATI
      LONGI=LONGIO+X/(RN*COS(LATI))
      S2=LONGI
      CALL DEGREE(D1,BMI,SE,S1)
      LA=D1+BMI*1E-2+SE*1E-4
      CALL DEGREE(D1,BMI,SE,S2)
      LO=D1+BMI*1E-2+SE*1E-4
      T=T+TRO
      TI=HOUR(T)
      T=T-TRO
      NM=N
      WRITE(*,10)DATE,TI,LA,LO,Z,MAG,PIMN,NM
   10 FORMAT(/,22X,'PARAMETERS OF THE HYPOC-
         ENTRE',
        //,20X,'DATE,' 10X,F15.4,/,20X,'ORIGIN TIME',
         3X,F15.4,/,20X,'LATITUDE',6X,F16.5,/,20X,'
         LONGITUDE',5X,F16.5,/,20X,'DEPTH' 9X,F15.4,
         /,20X,'MAGNITUDE',5X,F12.1,/,20X,' STAND-
         ARD ERROR',
         51X,'+/-',1X,F7.1,1X,'SEC.',/,20X,'USED
         STATIONS',3X,I6)
       RETURN
       END
C
C
C
       SUBROUTINE FUNCT(U,F)
       IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
       INTEGER STATN
       DIMENSION U(4),F(30)
```

```
      REAL LONGIO,LATIO,LONGI,LATI,MAG
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
         RM,PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,
         LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),TAO
         (30),
         STATD(31,2),BJ(30,4),DJ(3,4),XYZT(4),RESID
         (30),
         BJAC(30,5), STATN(30)
      COMMON/STAT/STATI(21,3),
         DEPTH(3),DELTA(104),BJC(3,104),FJC(104)
      COMMON/STA/STANA(21)
      CHARACTER*12 STANA
      DO 10 I=1,N
      DS=DSQRT((XB(I)-U(1))**2+(YB(I)-U(2))**
         2)
      CALL QIP(104,DS,DELTA,FJC,W)
      F(I)=W+U(4)
   10 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE CALPHI(PHI,F,DF)
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      INTEGER STATN
      REAL LONGIO,LATIO,LONGI,LATI,MAG
      DIMENSION F(30),DF(30)
```

```
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
         RM,PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,LA-
      TI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),
         TAO(30),STATD(31,2),BJ(30,4),DJ(3,4),
         XYZT(4),RESID(30),BJAC(30,5),STATN(30)
      PHI=0.
      DO 10 I=1,N
      DF(I)=TP(I)-F(I)
      RESID(I)=DF(I)
      PHI=PHI+DF(I)**2
   10 CONTINUE
      RETURN
      END
C
C
C
      SUBROUTINE CHECKS(KK)
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      INTEGER STATN
      REAL LONGIO,LATIO,LONGI,LATI,MAG
         DIMENSION DF(30)
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
         RM,PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,
         LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),
         TAO(30),STATD(31,2),BJ(30,4),DJ(3,4),
         XYZT(4),RESID(30),BJAC(30,5),STATN(30)
```

```
      CALL FUNCT(XYZT,TAO)
      CALL CALPHI(PHI,TAO,DF)
      KK=0
      DO 10 I=1,N
      IF(DABS(DF(I)).GE.3.0)THEN
      KK=KK+1
      STATN(I)=0
      END IF
   10 CONTINUE
      IF(KK.EQ.0)RETURN
      IF((N-KK).LT.5)THEN
      WRITE(*,20)
   20 FORMAT(/,10X',*****STATIONS LESS THAN
        5',/)
      RETURN
      END IF
      DO 30 NU=1,KK
      DO 25 I=1,N
      IF(STATN(I).EQ.0)THEN
      DO 22 J=I,N-1
      STATN(J)=STATN(J+1)
      XB(J)=XB(J+1)
      YB(J)=YB(J+1)
      ZB(J)=ZB(J+1)
      TP(J)=TP(J+1)
   22 CONTINUE
      END IF
   25 CONTINUE
   30 CONTINUE
      RETURN
      END
C
```

```
C
C
      SUBROUTINE MAQT(LMDO,UO,M)
      IMPLICIT DOUBLE PRECISION(A-H,L,P-Z)
      INTEGER STATN,S
      REAL LONGIO,LATIO,LONGI,LATI,LMDO,
         LMD,MAG
      DIMENSION UO(M),U(4),G(4),A(4.5),DF(30),TA
         (30)
      COMMON/NN/N,NJC
      COMMON/XYZ/X,Y,Z,T,DATE,TRO,DS,VP,RN,
         RM,PIMN
      COMMON/SLM/MAG,LONGIO,LATIO,LONGI,
         LATI
      COMMON/XBYB/XB(30),YB(30),ZB(30),TP(30),
         TAO(30),STATD(31,2),BJ(30,4),DJ(3,4),
         XYZT(4),RESID(30),BJAC(30,5).STATN(30)
      LMD=LMDO
      ITER=0
   10 LMD=LMD/10
      ITER=ITER+1
      CALL FUNCT(UO,TAO)
      CALL CALPHI(PHIO,TAO,DF)
      K=1
   20 UO(K)=UO(K)+1E-2
      CALL FUNCT(UO,TA)
      DO 25 J=1,N
      BJAC(J,K)=(TA(J)-TAO))*1E2
      UO(K)=UO(K)-1E-2
   25 CONTINUE
      K=K+1
```

```
      IF(K.LE.M) GO TO 20
      DO 30 I=1,N
30    BJAC(I,M+1)=DF(I)
      CALL MOND(BJAC,A,N,M)
      DO 40 I=1,M
      G(I)=A(I,M+1)
      A(I,I)=A(I,I)+LMD
40    CONTINUE
      CALL GAUS(A,M,1)
      DO 50 I=1,M
50    U(I)=UO(I)+A(I,M+1)
      CALL FUNCT(U,TAO)
      CALL CALPHI(PHI,TAO,DF)
      IF(PHI.GE.PHIO)GOTO 70
      PHIO=PHI
      DO 60 I=1,M
60    UO(I)=U(I)
      IF((ITER.LT.15).AND.((DSQRT(A(1,M+1)**2+A
        (2,M+1)
        **2)).GT.1.).AND.(A(M,M+1).GT.0.5))THEN
      GO TO 10
      ELSE
      GO TO 200
      END IF
70    GD=0.
      DO 80 I=1,M
80    GD=GD+G(I)*A(I,M+1)
      C=GD/(PHI-PHIO+2*GD)
      S=0
100   S=S+1
      DO 110 I=1,M
110   U(I)=UO(I)+C*A(I,M+1)
```

```
      CALL FUNCT(U,TAO)
      CALL CALPHI(PHI,TAO,DF)
      IF(PHI.GE.PHIO)GOTO 150
      PHIO=PHI
      DO 120 I=1,M
120   UO(I)=U(I)
      IF((ITER.LT.15).AND.((DSQRT(A(1,M+1)**2
     + +A(2,M+2)
     + **2)).GT.1.).AND.(A(M,M+1).GT.0.5))THEN
      GOTO 10
      ELSE
      GO TO 200
      END IF
150   IF(S.LT.7)THEN
      C=C*0.5
      GO TO 100
      END IF
200   PIMN=DSQRT(PHI/(N-3))
      RETURN
      END
```

## §5-9 应 用 实 例

例如，1976年4月25日沙城2.3级地震，有12台记录到，其观测数据如下：

| 台　号 | 台　名 | 到　时 |
|---|---|---|
| 2 | 沙　城 | 11-32-37.5 |
| 3 | 平　谷 | 11-32-38.2 |
| 4 | 马道峪 | 11-32-33.3 |
| 7 | 白家疃 | 11-32-28.7 |
| 5 | 龙泉寺 | 11-32-29.1 |
| 6 | 周口店 | 11-32-34.1 |

| | | |
|---|---|---|
| 8 | 桐　柏 | 11-32-37.7 |
| 17 | 涿　鹿 | 11-32-40.6 |
| 10 | 南山村 | 11-32-39.6 |
| 11 | 上房山 | 11-32-34.9 |
| 14 | 涞　源 | 11-32-45.3 |
| 13 | 张家口 | 11-32-48.1 |

下面是1976年4月25日沙城2.3级地震定位计算输出数据(包括原始数据及表格)：

* * * * * SWITCH LIST * * * * * (开关标号表)

1 1 1 1 1　　1 1 1 1 1　　1 1 1 1 1

* * * * * STATION NAME * * * * * (台站名及编号)

| | | | | | |
|---|---|---|---|---|---|
| 1 | Labagou | 2 | Shacheng | 3 | Pinggu |
| 4 | Madaoyu | 5 | Longquansi | 6 | Zhoukoudian |
| 7 | Baijiatan | 8 | Tongbai | 9 | Taisitun |
| 10 | Nansancun | 11 | Shangfangsan | 12 | Wenan |
| 13 | Zhangjiakou | 14 | LaiYuan | 15 | Wanxian |
| 16 | Huangbizuang | 17 | Zhuolu | 18 | Changli |
| 19 | Chengde | 20 | Lutai | 21 | Sanlihe |

(台站经纬度及高程)

| PHI | LAMBAD | HIGH |
|---|---|---|
| 40.535840 | 116.354490 | .400000 |
| 40.262500 | 115.312600 | .000000 |
| 40.060950 | 117.074070 | .070000 |
| 40.234100 | 116.334100 | .000000 |
| 40.022500 | 116.103000 | .200000 |
| 39.411750 | 115.552220 | .120000 |
| 40.021500 | 116.102570 | .040000 |
| 39.354860 | 116.442570 | .000000 |
| 40.323000 | 117.080000 | .000000 |
| 40.080000 | 117.150000 | .000000 |
| 39.393500 | 115.484200 | .204000 |
| 38.504900 | 116.271350 | .030000 |
| 40.494560 | 114.540000 | .835000 |
| 39.315600 | 114.580000 | .000000 |
| 38.515000 | 115.034700 | .100000 |

| | | |
|---|---|---|
| 38.145110 | 114.182480 | .140000 |
| 40.092630 | 115.075940 | .100000 |
| 39.452800 | 119.053400 | .200000 |
| 41.005500 | 117.545600 | .000000 |
| 39.223360 | 117.441550 | .000000 |
| 39.543000 | 116.194200 | .100000 |

~~DEPTH~~（走时表震源深度）

| | | |
|---|---|---|
| .00 | 10.00 | 16.00 |

~~DELTA~~ （走时表震中距）

| | | | | |
|---|---|---|---|---|
| .0000 | 11.1100 | 22.2200 | 33.3300 | 44.4400 |
| 55.5500 | 66.6600 | 77.7700 | 88.8800 | 99.9900 |
| 111.1000 | 122.2100 | 133.3200 | 144.4300 | 155.5400 |
| 166.6500 | 177.7600 | 188.8700 | 199.9800 | 211.0900 |
| 222.2000 | 233.3100 | 244.4200 | 255.5300 | 266.6400 |
| 277.7500 | 288.8600 | 299.9700 | 311.0800 | 322.1900 |
| 333.3000 | 344.4100 | 355.5200 | 366.6300 | 377.7400 |
| 388.8500 | 399.9600 | 411.0700 | 422.1800 | 433.2900 |
| 444.4000 | 455.5100 | 466.6200 | 477.7300 | 488.8400 |
| 499.9500 | 511.0600 | 522.1700 | 533.2800 | 544.3900 |
| 555.5000 | 566.6100 | 577.7200 | 588.8300 | 599.9400 |
| 611.0500 | 622.1600 | 633.2700 | 644.3800 | 655.4900 |
| 666.6000 | 677.7100 | 688.8200 | 699.9300 | 711.0400 |
| 722.1500 | 733.2600 | 744.3700 | 755.4800 | 766.5900 |
| 777.7000 | 788.8100 | 799.9200 | 811.0300 | 822.1400 |
| 833.2500 | 844.3600 | 855.4700 | 866.5800 | 877.6900 |
| 888.8000 | 899.9100 | 911.0200 | 922.1300 | 933.2400 |
| 944.3500 | 955.4600 | 966.5700 | 977.6800 | 988.7900 |
| 999.9000 | 1011.0100 | 1022.1200 | 1033.2300 | 1044.3400 |
| 1055.4500 | 1066.5600 | 1077.6700 | 1088.7800 | 1099.8900 |
| 1111.0000 | 1122.1100 | 1133.2200 | 1144.3300 | |

~~TRAVEL TIME~~ （三个不同震源深度的走时）

| | | |
|---|---|---|
| .0000 | 1.7500 | 2.8100 |
| 1.9500 | 2.6200 | 3.4200 |
| 3.9000 | 4.2700 | 4.8000 |
| 5.8500 | 6.1000 | 6.4900 |
| 7.8000 | 7.9900 | 8.2900 |
| 9.7500 | 9.9000 | 10.1400 |
| 11.6900 | 11.8300 | 12.0300 |
| 13.6400 | 13.7600 | 13.9300 |

| | | |
|---|---|---|
| 15.5900 | 15.6900 | 15.8400 |
| 17.5400 | 17.6300 | 17.7700 |
| 19.4900 | 19.5700 | 19.6900 |
| 21.4400 | 21.5100 | 21.6200 |
| 23.3900 | 23.4600 | 23.5600 |
| 25.3400 | 25.4000 | 25.4900 |
| 27.2900 | 27.3400 | 27.4300 |
| 29.2400 | 29.2900 | 29.3700 |
| 31.1900 | 31.2400 | 31.3100 |
| 33.1400 | 33.1800 | 33.2500 |
| 35.0800 | 35.1300 | 35.2000 |
| 37.0300 | 37.0700 | 37.1400 |
| 38.9300 | 39.0200 | 39.0800 |
| 40.9300 | 40.9700 | 41.0300 |
| 42.8800 | 42.9200 | 42.9700 |
| 44.8300 | 44.8600 | 44.9200 |
| 46.7800 | 46.8100 | 46.8600 |
| 48.7300 | 48.7600 | 48.8100 |
| 50.6800 | 50.7100 | 50.7500 |
| 52.6300 | 52.6600 | 52.7000 |
| 54.5800 | 54.6000 | 54.6500 |
| 56.5200 | 56.5500 | 56.5900 |
| 58.4700 | 58.5000 | 58.5400 |
| 60.4200 | 60.4500 | 60.4900 |
| 62.3700 | 62.4000 | 62.4400 |
| 64.3200 | 64.3400 | 64.3800 |
| 66.2700 | 66.2900 | 66.3300 |
| 68.2200 | 68.2400 | 68.2800 |
| 70.1700 | 70.1900 | 70.2200 |
| 72.1200 | 72.1400 | 72.1700 |
| 74.0700 | 74.0900 | 74.1200 |
| 76.0200 | 79.0400 | 76.0700 |
| 77.9600 | 77.9800 | 78.0200 |
| 79.9100 | 79.9300 | 79.9600 |
| 81.8600 | 81.8800 | 81.9100 |
| 83.8100 | 83.8300 | 83.8600 |
| 85.7600 | 85.7800 | 85.8100 |
| 87.7100 | 87.7300 | 87.7600 |
| 89.6600 | 89.6800 | 89.7000 |
| 91.6100 | 91.6300 | 91.6500 |
| 93.5600 | 93.5700 | 93.6000 |
| 95.5100 | 95.5200 | 95.5500 |
| 97.4600 | 97.4700 | 97.5000 |

| | | |
|---|---|---|
| 99.4100 | 99.4200 | 99.4400 |
| 101.3500 | 101.3700 | 101.3900 |
| 103.3000 | 103.3200 | 103.3400 |
| 105.2500 | 105.2700 | 105.2900 |
| 107.2000 | 107.2200 | 107.2400 |
| 109.1500 | 109.1600 | 109.1900 |
| 111.1000 | 111.1100 | 111.1400 |
| 113.0500 | 113.0600 | 113.0800 |
| 115.0000 | 115.0100 | 115.0300 |
| 116.9500 | 116.9600 | 116.9800 |
| 118.9000 | 118.9100 | 118.9300 |
| 120.8500 | 120.8600 | 120.8800 |
| 122.7900 | 122.8100 | 122.8300 |
| 124.7400 | 124.7600 | 124.7800 |
| 126.9600 | 126.7100 | 126.7200 |
| 128.6400 | 128.6500 | 128.6700 |
| 130.5900 | 130.6000 | 130.6200 |
| 132.5400 | 132.5500 | 132.5700 |
| 134.4900 | 134.5000 | 134.5200 |
| 136.4400 | 136.4500 | 136.4700 |
| 138.3900 | 138.4000 | 138.4200 |
| 140.3400 | 140.3500 | 140.3600 |
| 142.2900 | 142.3000 | 142.3100 |
| 144.2400 | 144.2500 | 144.2600 |
| 146.1800 | 146.1900 | 146.2100 |
| 148.1300 | 148.1400 | 148.1600 |
| 150.0800 | 150.0900 | 150.1100 |
| 152.0300 | 152.0400 | 152.0600 |
| 153.9800 | 153.9900 | 154.0100 |
| 155.9300 | 155.9400 | 155.9600 |
| 157.8800 | 157.8900 | 157.9000 |
| 159.8300 | 159.8400 | 159.8500 |
| 161.7800 | 161.7900 | 161.8000 |
| 163.7300 | 163.7400 | 163.7500 |
| 165.6800 | 165.6800 | 165.7000 |
| 167.6200 | 167.6300 | 167.6500 |
| 169.5700 | 169.5800 | 169.6000 |
| 171.5200 | 171.5300 | 171.5500 |
| 173.4700 | 173.4800 | 173.4900 |
| 175.4200 | 175.4300 | 175.4400 |
| 177.3700 | 177.3800 | 177.3900 |
| 179.3200 | 179.3300 | 179.3400 |
| 181.2700 | 181.2800 | 181.2900 |

| | | |
|---|---|---|
| 183.2200 | 183.2300 | 183.2400 |
| 185.2700 | 185.1700 | 185.1900 |
| 187.1200 | 187.1200 | 187.1400 |
| 189.0600 | 189.0700 | 189.0900 |
| 191.0100 | 191.0200 | 191.0300 |
| 192.9600 | 192.9700 | 192.9800 |
| 194.9100 | 194.9200 | 194.9300 |
| 196.8600 | 196.8700 | 196.8800 |
| 198.8100 | 198.8200 | 198.8300 |
| 200.7600 | 200.7700 | 200.7800 |

N=12(台站数) NJC=1(走时表选择)

| STATION NUMBER | ARRIVAL TIME |
|---|---|
| 1976.0 | 2.300 |
| 2.0 | 113237.500 |
| 3.0 | 113238.200 |
| 4.0 | 113233.300 |
| 21.0 | 113228.700 |
| 5.0 | 113229.100 |
| 6.0 | 113234.100 |
| 8.0 | 113237.700 |
| 17.0 | 113240.600 |
| 10.0 | 113239.600 |
| 11.0 | 113234.900 |
| 14.0 | 113245.300 |
| 13.0 | 113248.100 |

## PRELIBMINARY VALUE(初定值)

PARAMETERS OF THE HYPOCENTRE

| | |
|---|---|
| DATE | 1976.0425 |
| ORIGIN TIME | 113224.4450 |
| LATITUDE | 40.04039 |
| LONGITUDE | 116.13270 |
| DEPTH | 26.6224 |
| MAGNITUDE | 2.3 |
| STANDARD ERROR+/− | .0SEC. |
| USEDSTATIONS | 12 |

## MODIFIED VALUE(修定值)

### PARAMETERS OF THE HYPOCENTRE

| | |
|---|---|
| DATE | 1976.0425 |
| ORIGIN TIME | 113224.5433 |
| LATITUDE | 40.04218 |
| LONGITUDE | 116.13140 |
| DEPTH | 12.0000 |
| MAGNITUDE | 2.3 |
| STANDARD ERROR +/- | .7SEC. |
| USED STATIONS | 11 |

### CODE AND ARRIVAL TIME FOR RECORDING STATIONS

| STATION NAME(台站名) | | ARRIVAL TIME(到时) | RESIDUAL(残差) |
|---|---|---|---|
| 2 | Shacheng | 113237.5 | .3 |
| 3 | Pinggu | 113238.2 | .1 |
| 4 | NadaoYu | 113233.3 | .7 |
| 21 | Sanlihe | 113228.7 | .6 |
| 6 | Zhoukoudian | 113234.1 | .8 |
| 8 | Tongbai | 113237.7 | 1.0 |
| 17 | Zhuolu | 113240.6 | —.3 |
| 10 | Nansancun | 113239.6 | —.4 |
| 11 | Shangfangsan | 113234.9 | .2 |
| 14 | LaiYuan | 113245.3 | —.7 |
| 13 | Zhangjiakou | 113248.1 | —1.0 |

注：括号中的中文字是作者作为打印结果所加的注释.

# 第六章　层状介质地震体波走时的计算

研究地壳-上地幔详细结构的有效方法之一是利用人工震源激发地震波，一般说来，地震波在地下层状介质中传播时，地面可以观测到直达波、反射波、首波，在一定的条件下，还可以接收到由速度梯度层引起的回折波。

反射波、首波及回折波的形成，是和地壳及上地幔介质的分层结构分不开的。在人工地震测深方法中，要详细研究这些波的运动学及动力学特点，分析地壳及地幔的几何结构和速度分布。这一章，主要讨论水平及倾斜层状介质中各种地震体波的走时曲线的正演、反演问题。

## §6-1　水平层状介质反射波走时曲线

假定地壳模型为水平成层结构。其中各层的波速$V$可以是恒定的（常速层），或仅随深度作线性变化（速度梯度层）。如图6-1-1所示。

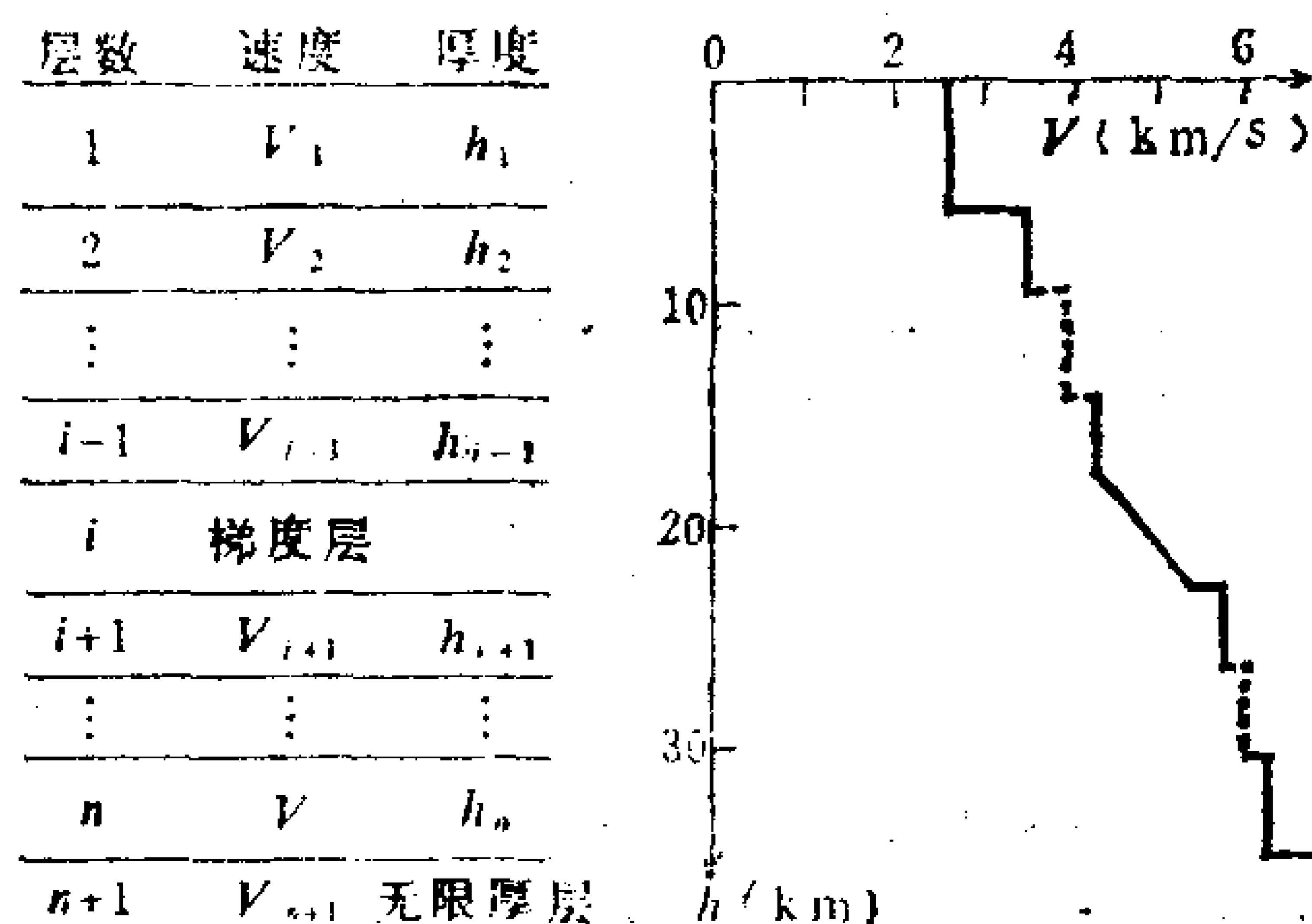



图 6-1-1

对于常速层，该层中速度梯度为零。若某层（例如第 $i$层)为变速层，则可定义该层的速度梯度为

$$a_i = \frac{V_{i+1} - V_{i-1}}{h_i}. \tag{6-1.1}$$

地震波由震源向下传播，在遇到地壳或上地幔分界面的波阻抗(即介质的密度与速度乘积 $\rho V_i$值)时，就会产生波的反射和折射。对于任意水平层的第 $n$ 个介面来说，反射波的走时与距离可用下述参数方程表示(参看图6-1-2)：

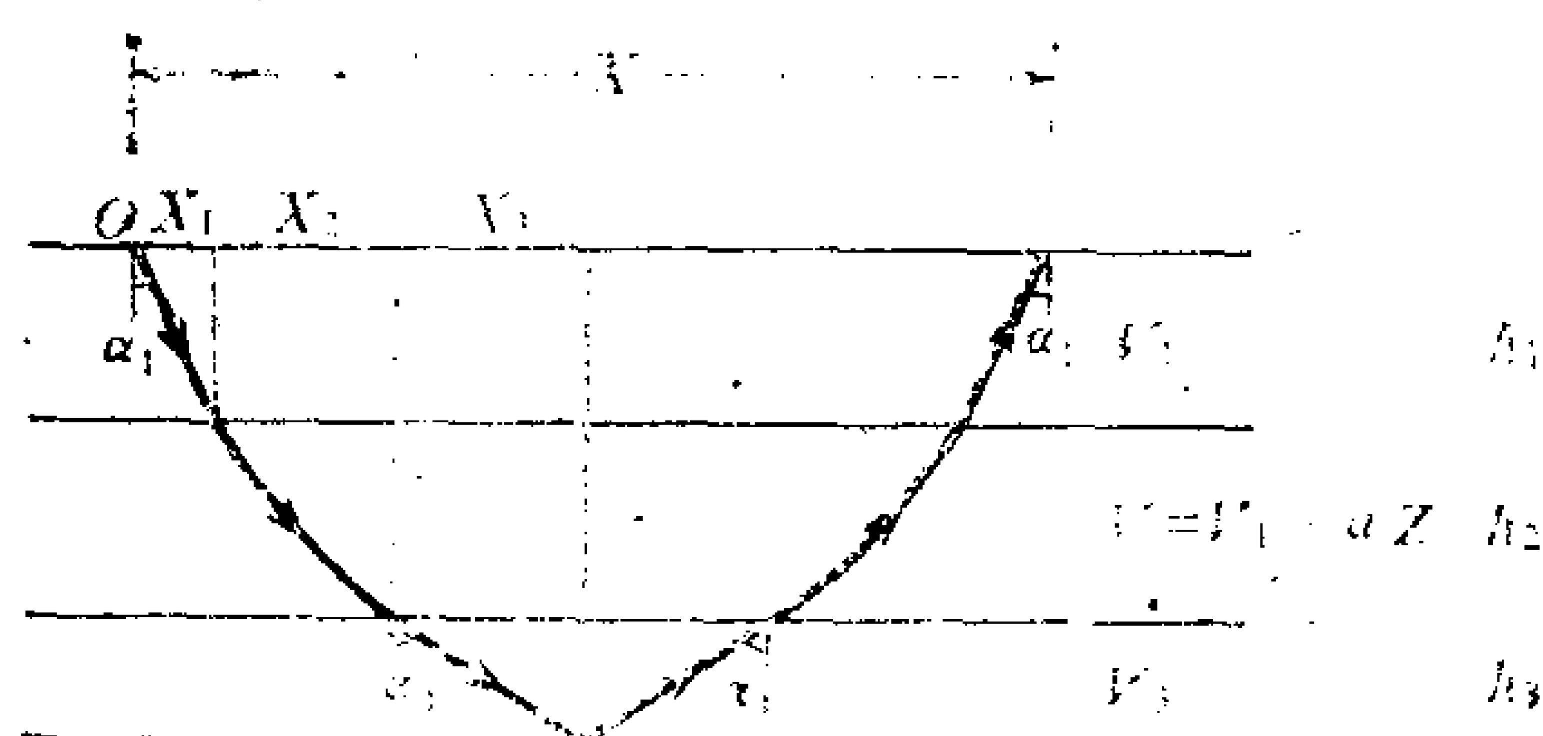


图 6-1-2

$$X = 2\sum_{i=1}^{n} x_i,$$

$$t = 2\sum_{i=1}^{n} t_i. \tag{6-1.2}$$

式中$X$表示震中距(即爆炸点到接收点的距离，又称炮检距)，$t$表示第$n$层界面反射波的走时，$x_i$及$t_i$分别为波在第$i$层中传播所经过的单程水平距离及走时。下面讨论$x_i$及$t_i$的计算方法。

由 Snell定律可知，若地面激发点地震射线以$\alpha_1$角入射，则通过各个速度不同的层时，在分界面上入射角有如下关系

$$\frac{\sin\alpha_1}{V_1} = \frac{\sin\alpha_2}{V_2} = \cdots = \frac{\sin\alpha_n}{V_n} = p, \tag{6-1.3}$$

或写为 $\sin\alpha_i = pV_i$, $(i = 1, 2, \cdots, n)$. (6-1.4)

式中 $p$称为射线参数。关系式(6-1.3)或(6-1.4)意味着，只要某一条入射线的初始入射角(或离源角)$\alpha_1$一经确定后，其传播路径也就被确定了。在以后传播过程中，尽管因层速度不同使射线角度有变化，但其射线参数$p$仍不变化。

由图6-1-2可见，射线在常速层(设为第 $i$ 层)中经过的水平距离及走时为

$$x_i = h_i \mathrm{tg}\alpha_i = h_i \frac{\sin\alpha_i}{\cos\alpha_i} = h_i \frac{pV_i}{\sqrt{1-p^2V_i^2}},$$
$$t_i = \frac{h_i}{V_i\cos\alpha_i} = \frac{h_i}{V_i\sqrt{1-p^2V_i^2}}. \qquad (6\text{-}1.5)$$

式中$h_i$及$V_i$分别为第 $i$层介质的厚度及波速，$\alpha_i$是射线在$i$层上界面的入射角。

若第$i$层为速度梯度层时，可按如下方法进行计算。

如图6-1-3所示，我们假定梯度层顶部的波速与 $i-1$ 层相等，为 $V_{i-1}$；梯度层底部的波速与 $i+1$层相等，为 $V_{i+1}$。如果在速度梯度层顶部(或底部)与上(下)层的速度不连续，那么就要假定梯度层的顶部(或底部)存在着一个速度恒定的层，这层的速度与梯度层的顶部(或底部)速度相等，但厚度为零。于是，我们很容易由上层速度 $V_{i-1}$和下层速度 $V_{i+1}$ 求得梯度层中的速度梯度

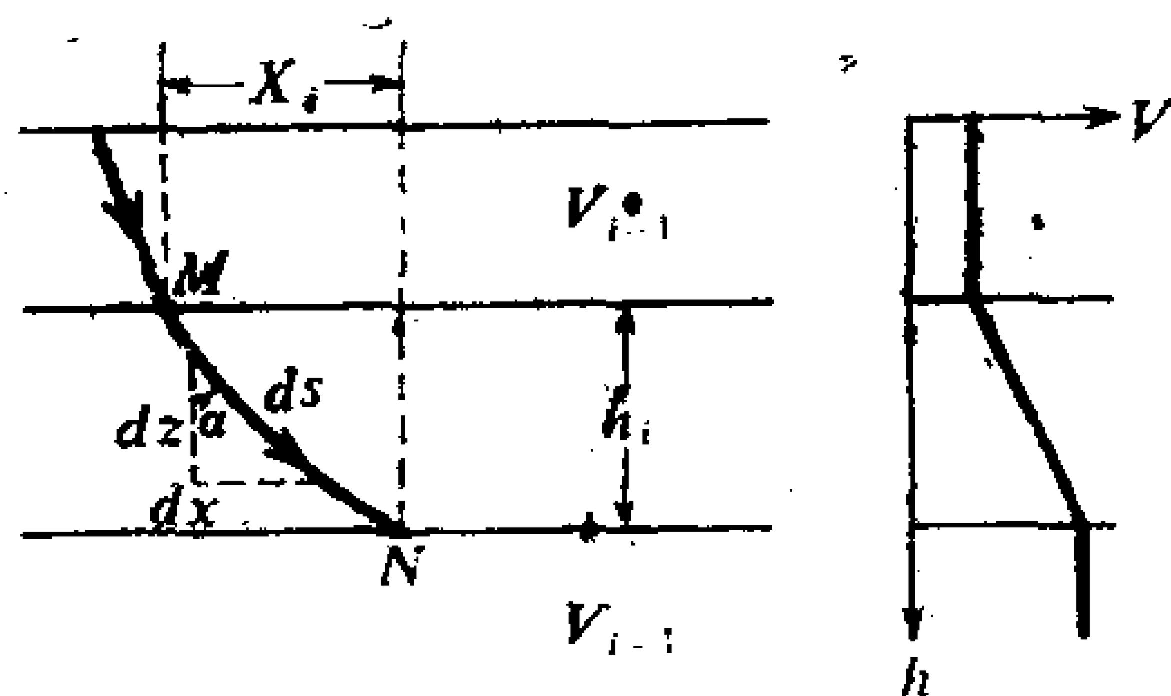


图 6-1-3

$$a_i = \frac{V_{i+1} - V_{i-1}}{h_i}. \qquad (6\text{-}1.6)$$

则射线从 $M$点穿透到 $N$点经过的水平距离为

$$x_i = \int_0^{h_i} \mathrm{tg}\alpha dz$$
$$= \int_0^{h_i} \frac{pV}{\sqrt{1-p^2V^2}} dz$$

$$= \frac{1}{a_i}\int_{V_{i-1}}^{V_{i+1}} \frac{pV}{\sqrt{1-p^2V^2}} dV$$

$$= \frac{1}{a_i p}\left(\sqrt{1-p^2V_{i-1}^2} - \sqrt{1-p^2V_{i+1}^2}\right)$$

$$= \frac{1}{a_i p}(\cos\alpha_{i-1} - \cos\alpha_{i+1}).$$

相应的走时为

$$t_i = \int_0^{h_i} \frac{dz}{V\cos\alpha}$$

$$= \frac{1}{a_i}\int_{V_{i-1}}^{V_{i+1}} \frac{dV}{V\sqrt{1-p^2V^2}}$$

$$= \frac{1}{a_i}\left(\ln\frac{pV_{i+1}}{1+\sqrt{1-p^2V_{i+1}^2}} - \ln\frac{pV_{i-1}}{1+\sqrt{1-p^2V_{i-1}^2}}\right)$$

$$= \frac{1}{a_i}\left(\ln\frac{\sin\alpha_{i+1}}{1+\cos\alpha_{i+1}} - \ln\frac{\sin\alpha_{i-1}}{1+\cos\alpha_{i-1}}\right)$$

$$= \frac{1}{a_i}\ln\left(\frac{\mathrm{tg}\frac{\alpha_{i+1}}{2}}{\mathrm{tg}\frac{\alpha_{i-1}}{2}}\right).$$

综上得速度梯度层反射波传播的参数方程为

$$\left.\begin{aligned} x_i &= \frac{1}{a_i p}(\cos\alpha_{i-1} - \cos\alpha_{i+1}), \\ t_i &= \frac{1}{a_i}\ln\left(\frac{\mathrm{tg}\frac{\alpha_{i+1}}{2}}{\mathrm{tg}\frac{\alpha_{i-1}}{2}}\right). \end{aligned}\right\} \quad (6\text{-}1.7)$$

若将上式改写为以射线参数 $p$表示的参数方程，则为

$$\left.\begin{aligned} x_i &= \frac{1}{a_i p}[(1-p^2V_{i-1}^2)^{½} - (1-p^2V_{i+1}^2)^{½}], \\ t_i &= \frac{1}{a_i}\left[\cosh^{-1}\left(\frac{1}{pV_{i-1}}\right) - \cosh^{-1}\left(\frac{1}{pV_{i+1}}\right)\right]. \end{aligned}\right\} \quad (6\text{-}1.8)$$

由(6-1.5)，(6-1.7)及(6-1.8)式可见，反射波走时方程是一组以射线参数$p$或入射角 $\alpha_1$为参量的参数方程。一般说来，

要计算走时，先要给定某一射线的初始入射角$\alpha_1$，由(6-1.3)式求得射线参数$p$。再由(6-1.5)或(6-1.8)式计算通过各层的距离$x_i$及走时$t_i$。最后用(6-1.2)式计算总的震中距及走时。但是，按给定的入射角$\alpha_1$去计算走时曲线震中距分布是不均匀的，很难与实际观测的走时曲线对比，这对于反演计算是不方便的。实际计算是把这个次序颠倒过来，即先给定震中距$X$，去反求与此$X$相对应的射线的入射角$\alpha_1$或射线参数$p$。根据此$\alpha_1$或$p$值再求相应的走时$t$。

为求出给定震中距$X$相应的射线入射角$\alpha_1$(或射线参数$p$)，可以构造一个非线性方程

$$F(\alpha)=X-f(h_1,h_2,\cdots,h_h,V_1,V_2,\cdots,V_n,\alpha)=0. \tag{6-1.9}$$

其中，观测点震中距$X$及各层厚度、速度等参数均为已知数。函数$f$表示理论计算震中距，它可以由前述几个计算单层水平距离$x_i$的公式组合构成。对应于非线性方程$F(\alpha)=0$的根就是待求的入射角$\alpha_1$。由于这个函数比较复杂，只能用下述数值方法求解。

## §6-2 非线性方程求根法

求非线性方程$F(\alpha)=0$的根，一般是采用某种迭代解法。即从某一初值开始，利用迭代格式构造一组近似值序列，逐步逼近于所求的解$\alpha_1$值。

迭代格式分两步执行，第一步是搜索根所存在的范围，为此，由经验的办法从一起始的入射角$\alpha_0$按步长$\Delta\alpha$使其逐次增大，即依次计算$F(\alpha_0+i\Delta\alpha)$值($i=0, 1, 2, \cdots\cdots$)。一旦相邻两点的函数值反号，即$F(\alpha_0+i\Delta\alpha)$与$F(\alpha_0+(i+1)\Delta\alpha)$值异号时，则知待求的根$\alpha_1$一定位于$(\alpha_0+i\Delta\alpha, \alpha_0+(i+1)\Delta\alpha)$区间之中。

对给定区间求非线性方程的根常用方法是线性插值法及二次插值法，它们的计算效率较高，程序编制亦较简单。

## (一) 线性插值法

又称弦位法。设已知方程

$$F(\alpha)=0 \qquad (6\text{-}2.1)$$

的根在区间$(\alpha_1,\ \alpha_2)$内，它们在两端相应的函数值为$F_1$及$F_2$，即

$$F_1=F(\alpha_1),\qquad F_2=F(\alpha_2).$$

且$F_1$及$F_2$值异号。通过这两个点

$$(\alpha_1,\ F_1)\ \text{及}\ (\alpha_2,\ F_2),$$

构造一个线性函数(见图6-2-1)

$$L(\alpha)=a\alpha+b. \qquad (6\text{-}2.2)$$

由图不难定出此线性函数为

$$L(\alpha)=F_2+\frac{F_2-F_1}{\alpha_2-\alpha_1}(\alpha-\alpha_2). \qquad (6\text{-}2.3)$$

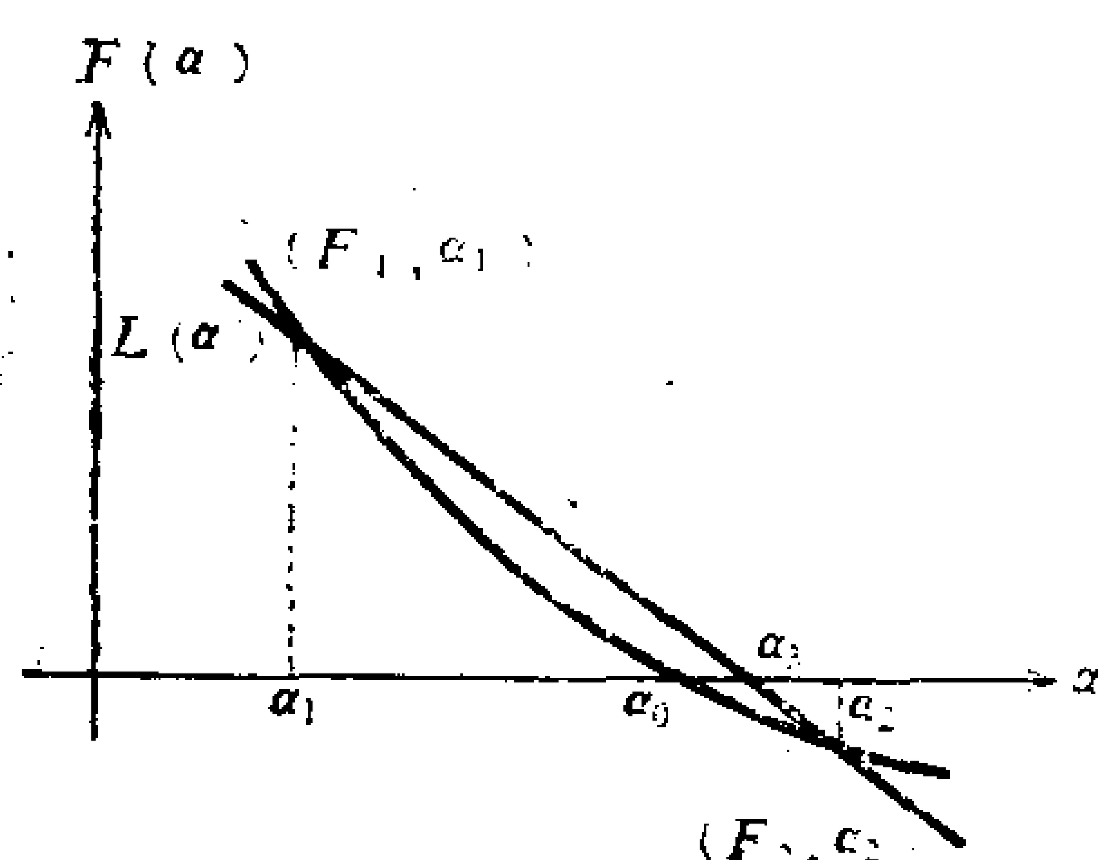



图 6-2-1

其意义是用一直线(弦线)逼近曲线。在根的近似值$\alpha_1$、$\alpha_2$附近，以线性函数$L(\alpha)$来近似函数$F(\alpha)$，以$L(\alpha)$的零点

$$\alpha_3=\alpha_2-F_2\frac{\alpha_2-\alpha_1}{F_2-F_1}, \qquad (6\text{-}2.4)$$

作为根的一个新的近似值。几何上就是以弦线与$\alpha$轴的交点$\alpha^3$来进一步逼近曲线与$\alpha$轴的交点$\alpha_0$(根)。

预期$\alpha_3$比$\alpha_1$及$\alpha_2$更接近于解$\alpha_0$，且$F(\alpha_3)$与$F(\alpha_1)$异号，于是将新的近似值$\alpha_3$代替$\alpha_2$，以$(\alpha_1,\ \alpha_3)$为新的求根区间。重复上述过程，又得到$\alpha_4$，如此反复执行，直到充分逼近解$\alpha_0$时为止。

用弦位法求根的计算步骤如下:

(1) 准备：根据搜索法选定初始近似值$\alpha_1$，$\alpha_2$，计算相应的函数值$F_1$及$F_2$。(注意，$F_1$与$F_2$异号)

（2）迭代：按公式(6-2.4)迭代一次，得到新的近似值$\alpha_3$，并计算$F_3 = F(\alpha_3)$。

（3）控制：计算限差$\delta = |\alpha_3 - \alpha_2|$，如果满足$\delta \leqslant \varepsilon$（$\varepsilon$是允许误差，事先给定），或者满足$|F_3| \approx 0$，则认为过程收敛而终止迭代。以$\alpha_3$作为所求的解，否则执行下一步。

（4）迭代准备：判断$F_3$与$F_2$及$F_1$同异号的关系。如$F_3$与$F_1$异号，则以$(\alpha_3, F_3)$点代替$(\alpha_2, F_2)$点。否则以$(\alpha_3, F_3)$点代替$(\alpha_1, F_1)$点。而后转入步骤2继续迭代。

## （二）二次插值法

又称Müller法。设已知非线性方程

$$F(\alpha) = 0,$$

其三个近似解为$\alpha_1$，$\alpha_2$，$\alpha_3$（都在求解区间内）。相应的$F(\alpha)$函数值是$F_1$，$F_2$，$F_3$。二次插值法的基本思想是，通过三个近似值点$(\alpha_1, F_1)$，$(\alpha_2, F_2)$及$(\alpha_3, F_3)$构造二次函数

$$L(\alpha) = a\alpha^2 + b\alpha + c \quad . \tag{6-2.5}$$

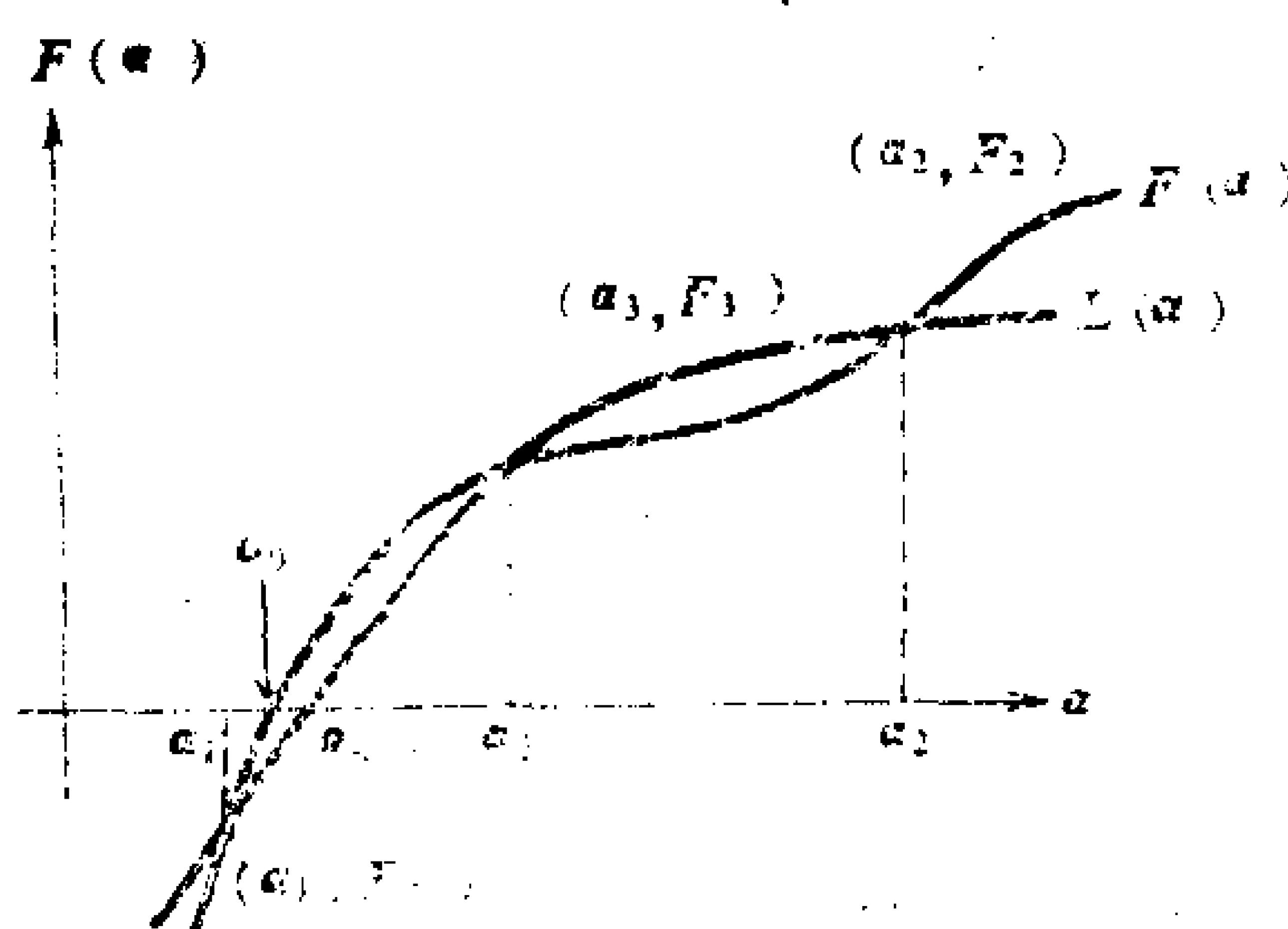



图 6-2-2

其图形即为通过上述三点的一条抛物线，如图6-2-2所示。

在近似值附近就以此二次函数$L(\alpha)$来逼近非线性函数$F(\alpha)$，并以此二次函数的两个零点之一作为解的进一步近似。

利用 Lagrange 插值公式，作出通过上述三点的二次函数

$$L(\alpha) = \frac{(\alpha - \alpha_2)(\alpha - \alpha_3)}{(\alpha_1 - \alpha_2)(\alpha_1 - \alpha_3)} F_1 + \frac{(\alpha - \alpha_1)(\alpha - \alpha_3)}{(\alpha_2 - \alpha_1)(\alpha_2 - \alpha_3)} F_2 + \frac{(\alpha - \alpha_1)(\alpha - \alpha_2)}{(\alpha_3 - \alpha_1)(\alpha_3 - \alpha_2)} F_3 \, . \tag{6-2.6}$$

为了计算方便，引入新的变量

$$\lambda=\frac{\alpha-\alpha_3}{\alpha_3-\alpha_2}, \tag{6-2.7}$$

且令

$$\lambda_3=\frac{\alpha_3-\alpha_2}{\alpha_2-\alpha_1},$$

$$\delta_3=1+\lambda_3. \tag{6-2.8}$$

于是，(6-2.6)式便改写为

$$L(\lambda)=a\lambda^2+b\lambda+c. \tag{6-2.9}$$

式中各常数为

$$a=F_1\lambda_3^2-F_2\lambda_3\delta_3+F_3\lambda_3,$$

$$b=F_1\lambda_3^2-F_2\delta_3^2+F_3(\lambda_3+\delta_3),$$

$$c=F_3\delta_3.$$

二次函数(6-2.9)式$L(\lambda)$存在两个根

$$\lambda=\frac{-2c}{b\pm\sqrt{b^2-4ac}}. \tag{6-2.10}$$

取其模小的一个（即分母中的正负号应取使分母的模值为大的那个），记为$\lambda_4$。由$\lambda$的定义得

$$\alpha_4=\alpha_3+\lambda_4(\alpha_3-\alpha_2), \tag{6-2.11}$$

作为一个新的近似值。然后以$\alpha_2$，$\alpha_3$，$\alpha_4$代替原来的$\alpha_1$，$\alpha_2$及$\alpha_3$三点，重复上述过程得到$\alpha_5$。依次迭代，直到近似值充分接近解答为止。具体计算步骤如下：

（1）准备：根据所选定的区间确定三个初始近似值$\alpha_1$，$\alpha_2$，$\alpha_3$，计算相应的函数值$F_1$，$F_2$，$F_3$，并计算

$$\lambda_3=\frac{\alpha_3-\alpha_2}{\alpha_2-\alpha_1}.$$

（2）迭代：计算

$$\delta_3=1+\lambda_3,$$

$$a=F_1\lambda_3^2-F_2\lambda_3\delta_3+F_3\lambda_3,$$

$$b = F_1\lambda_3^2 - F_2\delta_3^2 + F_3(\lambda_3 + \delta_3),$$

$$c = F_3\delta_3 \quad ,$$

$$\lambda_4 = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}}.$$

取分母模值为大的一个，如遇$F_1 = F_2 = F_3$或$\alpha_3 = \alpha_1$等异常情况，则取$\lambda_4 = 1$。

于是得到新的近似值

$$\alpha_4 = \alpha_3 + \lambda_4(\alpha_3 - \alpha_2).$$

计算$F_4 = F(\alpha_4)$。如果$F_4$不满足$|F_4|/|F_3| \leqslant M$（$M$是给定的常数），则缩小$\lambda_4$，直到这一条件满足为止。

（3）控制：如果$\alpha_4$满足

$$\delta = |\alpha_4 - \alpha_3| \leqslant \varepsilon, \quad (\varepsilon\text{事先给定}),$$

或

$$|F_4| \approx 0.$$

则认为过程收敛，终止迭代，以$\alpha_4$作为所求的解，否则执行下一步。

（4）迭代准备：如果迭代次数超过某个上界，则认为过程不收敛，停止迭代，计算失效。否则，以$\alpha_2$，$\alpha_3$，$\alpha_4$，$F_2$，$F_3$，$F_4$，$\lambda_4$分别代替$\alpha_1$，$\alpha_2$，$\alpha_3$，$F_1$，$F_2$，$F_3$，$\lambda_3$，转入步骤2继续迭代计算。

实际计算表明，求非线性方程$F(\alpha) = 0$的根时，上述两种方法都是有效的。一般说来，线性插值法程序较简单，但二次插值法有较高的效率。这两种方法的收敛速度都与初始值的选择有较大的关系，因此，用经验方法确定初值的工作很重要。

由(6-1.9)式可见，对非线性方程$P(\alpha)$求得初始入射角$\alpha_1$后，即可对相应的震中距X求得走时$t$。由某一介面产生的反射波，其分布范围是很广的。一般把反射角小于临界角的反射波称为窄角度反射波，大于临界角的称为宽角度反射波。我们只需要对一系列点给出震中距，就可以获得走时曲线X-$t$。

## §6-3 回折波及首波理论走时的计算

层状介质除产生反射波外，在一定的条件下还产生另外两类波，即回折波和首波。

### （一）回 折 波

当射线进入速度梯度层就会发生弯曲。如果速度随深度增大（速度梯度$a_i=(U_{i+1}-U_{i-1})/h_i>0$），射线将向上弯曲。当射线一直弯曲到临界角 90° 时，它将沿轴对称方向折回到地面。这就是所谓的回折波。如图6-3-1示。

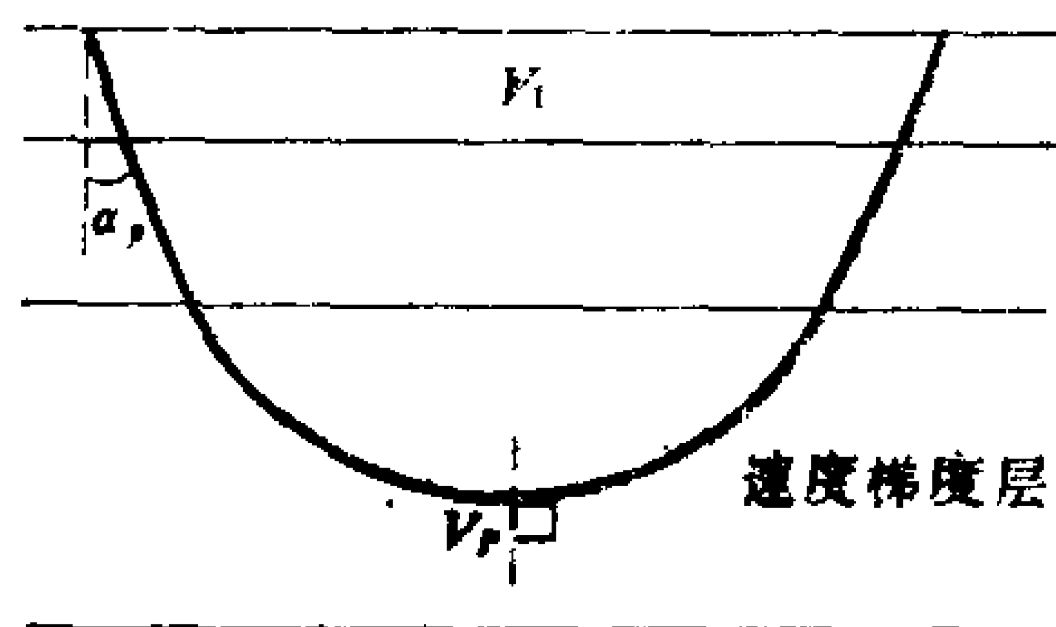



图 6-8-1

形成回折波的基本条件是必须存在正的速度梯度层，且梯度层中有速度大于以上各层中最大，当入射波初始入射角$\alpha_P$满足关系式

$$\sin\alpha_P=\frac{V_1}{V_P} \qquad (6-3.1)$$

时，就形成回折波。其中$V_P$是射线最深点（即回折点）处的波速。

根据速度梯度层的速度分布特征，回折波的走时曲线又分为顺进及逆进两种情况。

图6-3-2速度剖面中，速度梯度层的梯度值$a_i$较小(*a*)，回折波在地面存在的范围为*A*、*B*两点之间(*b*)。它们到达*A*、*B*的走时*T*是随震中距*X*增加而增加的，而且较深处回折的射线有着更远的距离，故走时曲线(*c*)上表现为顺进的。

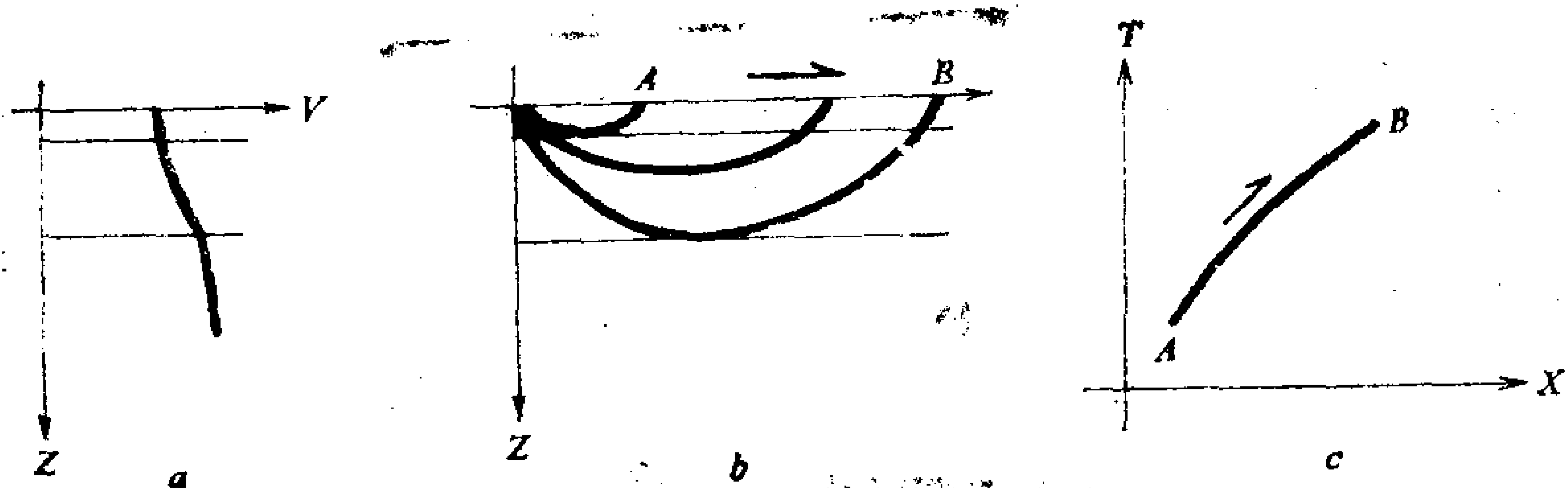



图 6-3-2

图6-3-3速度剖面(a)表示另一种情况，这里速度梯度层的梯度值$a_i$很大，以致回折波射线强烈弯曲。回折波在地面存在的范围为B、C两点之间(b)，它们到达B、C的走时T是随震中距X减小而减小的。在走时曲线(c)上表现为BC段。这时，由较深处回折的射线反而有较近的距离。故走时曲线(c)上BC段表现为逆进的。

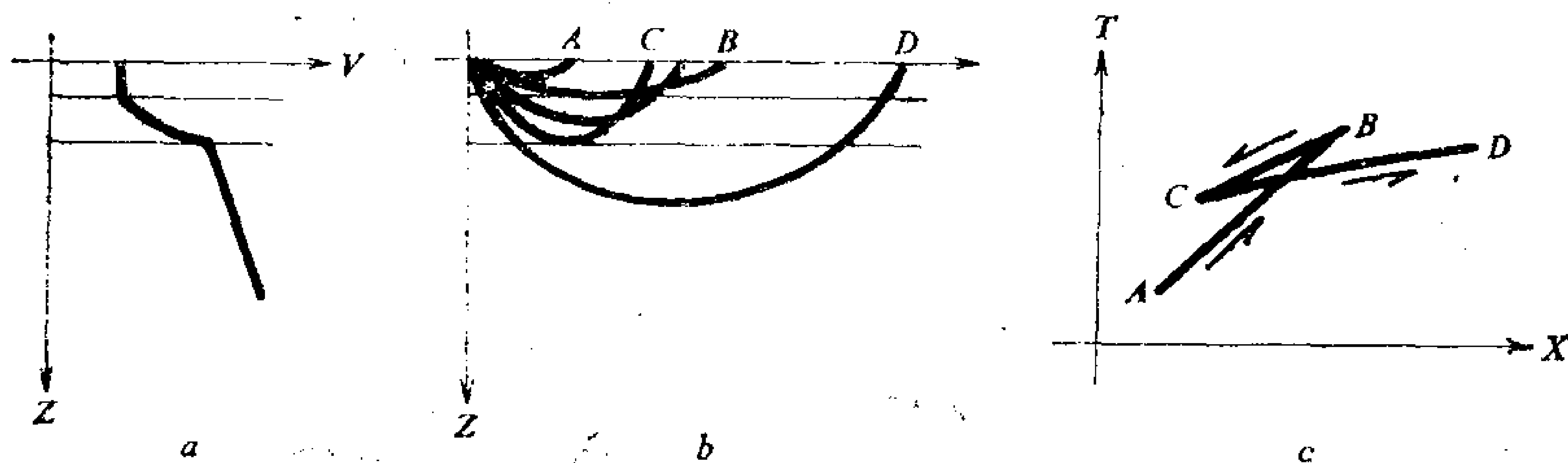



图 6-3-3

由于实际介质中速度梯度层都具有一定厚度，回折波存在范围是由该层上界面及下界面返回的两条射线所限定的。这两条射线的入射角分别满足关系式

$$\sin\alpha_1 = \frac{V_1}{V_{i-1}},$$

$$\sin\alpha_2 = \frac{V_1}{V_{i+1}}. \tag{6-3.2}$$

式中$V_{i-1}$为梯度层上界面的波速，$V_{i+1}$是梯度下界面的波速。

对于第 $n$层介质中回折波，震中距$X$及走时 $t$仍按(6-1.2)式计算。其中，从第一层到$n-1$层中的$x_i$及$t_i$的计算公式，仍归采用(6-1.5)、(6-1.7)或(6-1.8)式。仅对于产生回折波的第 $n$层介质，因为$\alpha_{n+1}=90°$，故(6-1.7)式可以简化为以下公式

$$x_n=\frac{1}{a_n p}\cos\alpha_{n-1},$$

$$t_n=\frac{1}{a_n}\ln\mathrm{ctg}\frac{\alpha_{n-1}}{2}. \tag{6-3.3}$$

$x_n$及 $t_n$分别表示梯度层内回折波所经过的水平距离及走时。$a_n$为第$n$层介质中的速度梯度。

## （二）首　　波

当某一界面下层介质的波速大于上层介质的波速，且震中距大于临界距离时，就可以在地面接收到来自该界面的首波，如图6-3-4所示。

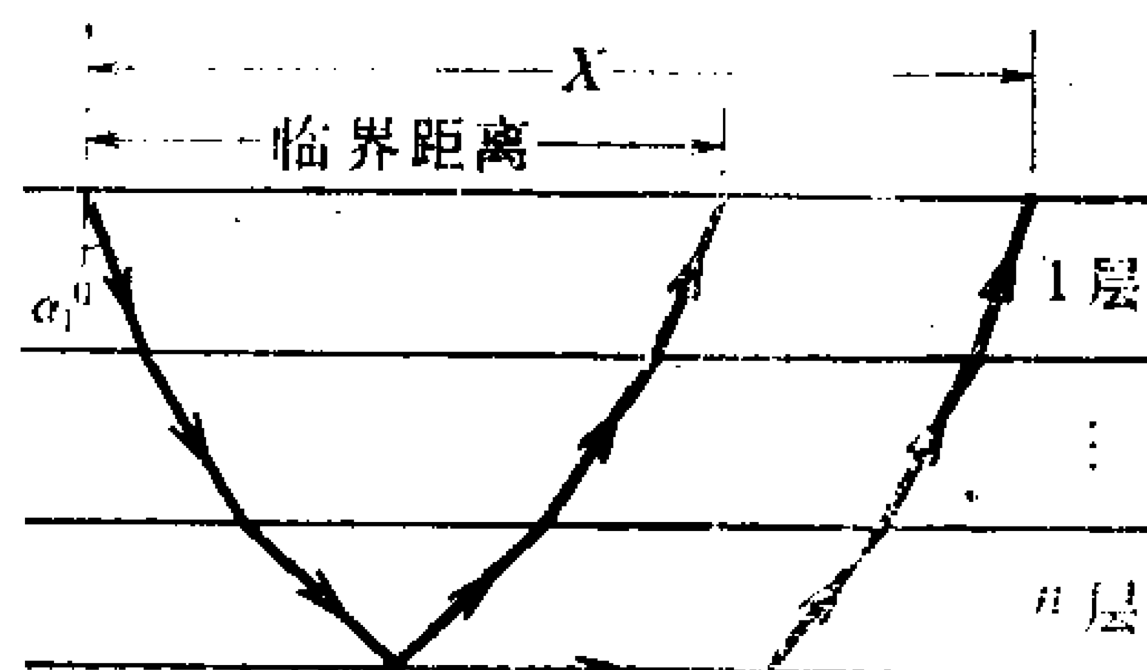



图 6-3-4

设首波在地面距震源 $X$处被接收，则折射波的走时可按下式计算

$$t_首=2\sum_{i=1}^{n}t_i^0+\frac{X-2\sum\limits_{i=1}^{n}x_i^0}{V_{n+1}}. \tag{6-3.4}$$

式中$t_i^0$及$x_i^0$分别为临界条件下，首波在第$i$层介质中穿透的单程走时及水平距离。

由于首波是在$n$层底界面(即$n+1$层顶界面)沿界面滑行的，所以存在以下关系

$$\frac{\sin\alpha_1^0}{V_1}=\frac{\sin\alpha_2^0}{V_2}=\cdots\cdots=\frac{\sin\alpha_n^0}{V_n}=\frac{\sin 90^\circ}{V_{n+1}}=\frac{1}{V_{n+1}}=p^0.$$

$\alpha_1^0$，$\alpha_2^0$，…$\alpha_n^0$为各层介质中相应于临界面角射线的入射角。地面初始入射角$\alpha_1^0$可以由下式计算

$$\sin\alpha_1^0=\frac{V_1}{V_{n+1}}. \tag{6-3.5}$$

有了初始入射角 $\alpha_1^0$（或射线参数 $p^0$），则以此代入(6-1.5)、(6-1.7)或(6-1.8)式，分别计算出单层走时$t_i^0$及距离$x_i^0$。

如果地壳模型各层介质都是匀速层而无速度梯度层时，(6-3.4)式可以简化为

$$t_{首}=2\sum_{i=1}^{n}\frac{h_i}{V_i}\cos\alpha_i^0+\frac{X}{V_{n+1}}. \tag{6-3.6}$$

中式 $\alpha_i^0=\sin^{-1}\left(\frac{V_i}{V_{n+1}}\right)$.

## §6-4 倾斜层状介质走时曲线的计算

这一节，讨论任意多层，但界面具有不同倾角的倾斜层状介质反射波及折射波走时曲线的计算。先讨论反射波问题。

如图6-4-1所示，若已知各层在原点$O$处的垂直厚度为$Z_1$，$Z_2$,…，$Z_n$；各层界面的倾角为$\phi_1$，$\phi_2$，…，$\phi_n$；而各层的速度为$V_1$，$V_2$，…，$V_n$。

为了计算方便，我们约定：1)倾角 $\phi_i(i=1,2,\cdots n)$相对于水平基线顺时针方向为正，反时针方向为负；2)所有的水平距离$\xi$相对于原点在右为正，在左为负；垂直距离Z向下为正；3)入射角$\alpha$及透射角$\beta$相对界面的法线方向反时针旋转为正，顺时针方向为负。

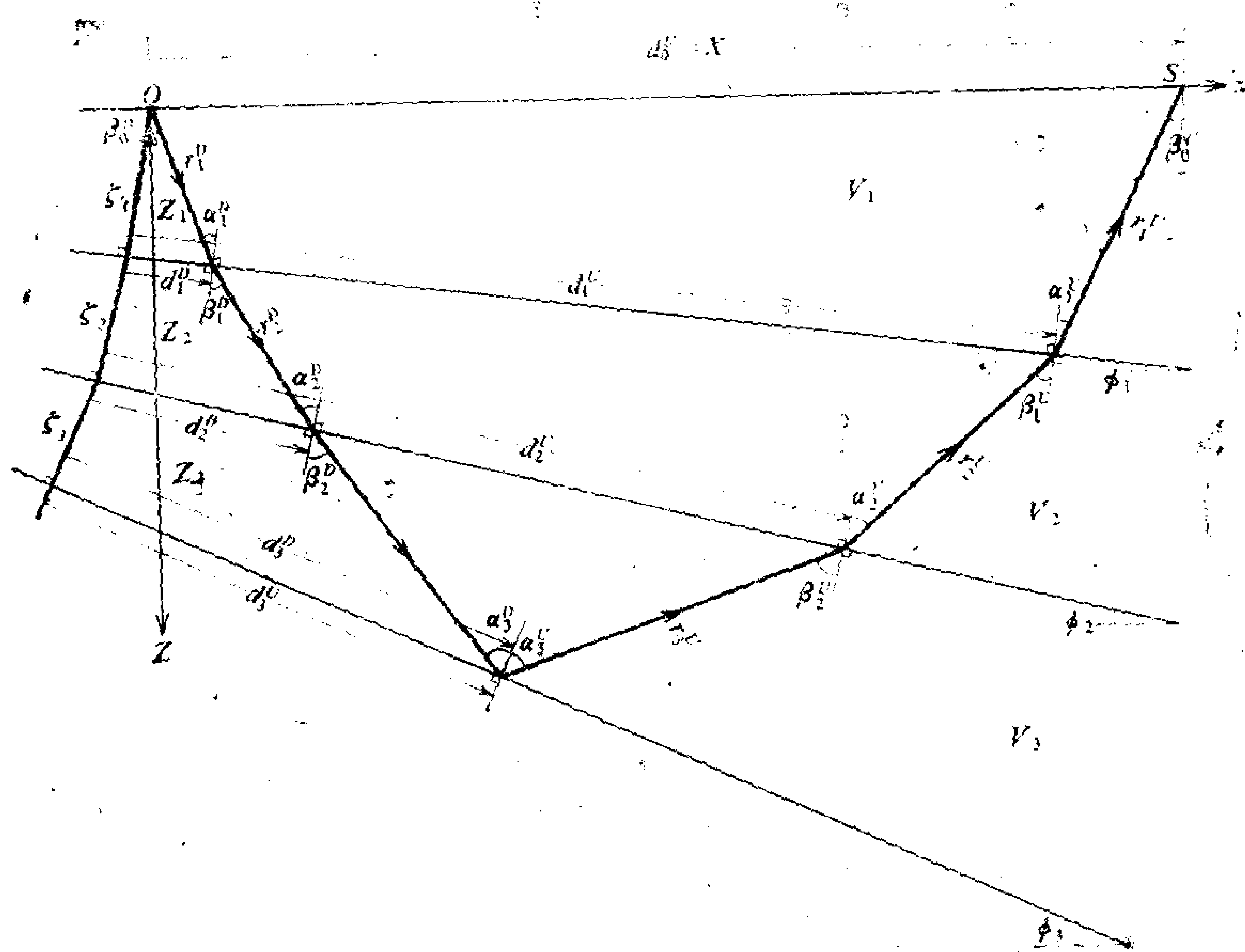


图 6-4-1

图6-4-1中各变量的意义是：$\zeta_i$为各层层间的垂直距离；$r_i^D$为各层地震入射线的路径（D表示下行，下同）；$r_i^U$为各层间地震出射线的路径（U表示上行，下同）；$d_i^D$为层间垂线和入射线$r_i^D$在界面上交点的距离；$d_i^U$为层间垂线$\zeta_i$与出射线$r_i^U$在界面上交点的距离。

根据Snell定律，可以求出各界面射线的入射角$\alpha_i^D$，$\alpha_i^U$及出射角$\beta_i^D$，$\beta_i^U$。这可由下列递推公式得出：

下行　$\alpha_i^D=\beta_{i-1}^D+\phi_i-\phi_{i-1}$，$\beta_i^D=\sin^{-1}\left(\sin\alpha_i^D\dfrac{V_{i+1}}{V_i}\right)$，

上行　$\beta_{i-1}^U=\alpha_i^U+\phi_i-\phi_{i-1}$，$\alpha_{i-1}^U=\sin^{-1}\left(\sin\beta_{i-1}^U\dfrac{V_{i-1}}{V_i}\right)$，

$$i=1,\ 2,\ \cdots,\ n. \qquad (6\text{-}4.1)$$

层面间的垂直距离$\zeta_i$是根据“垂直厚度”$Z_i$计算的，它们可

根据以下递推公式依次从上到下计算

$$\zeta_i = [Z_i - b_{i-1}(\mathrm{tg}\phi_i - \mathrm{tg}\phi_{i-1})]\cos\phi_i. \qquad (6\text{-}4.2)$$

$$b_i = b_{i-1} + \zeta_i \sin\phi_i, \quad 且 \quad b_0 \equiv 0. \qquad (6\text{-}4.2')$$

$$i = 1, 2, \cdots, n.$$

下面是求层间射线路径$r_i^D$，$r_i^U$。先求下行波，如图6-4-2示，可得出递推式为

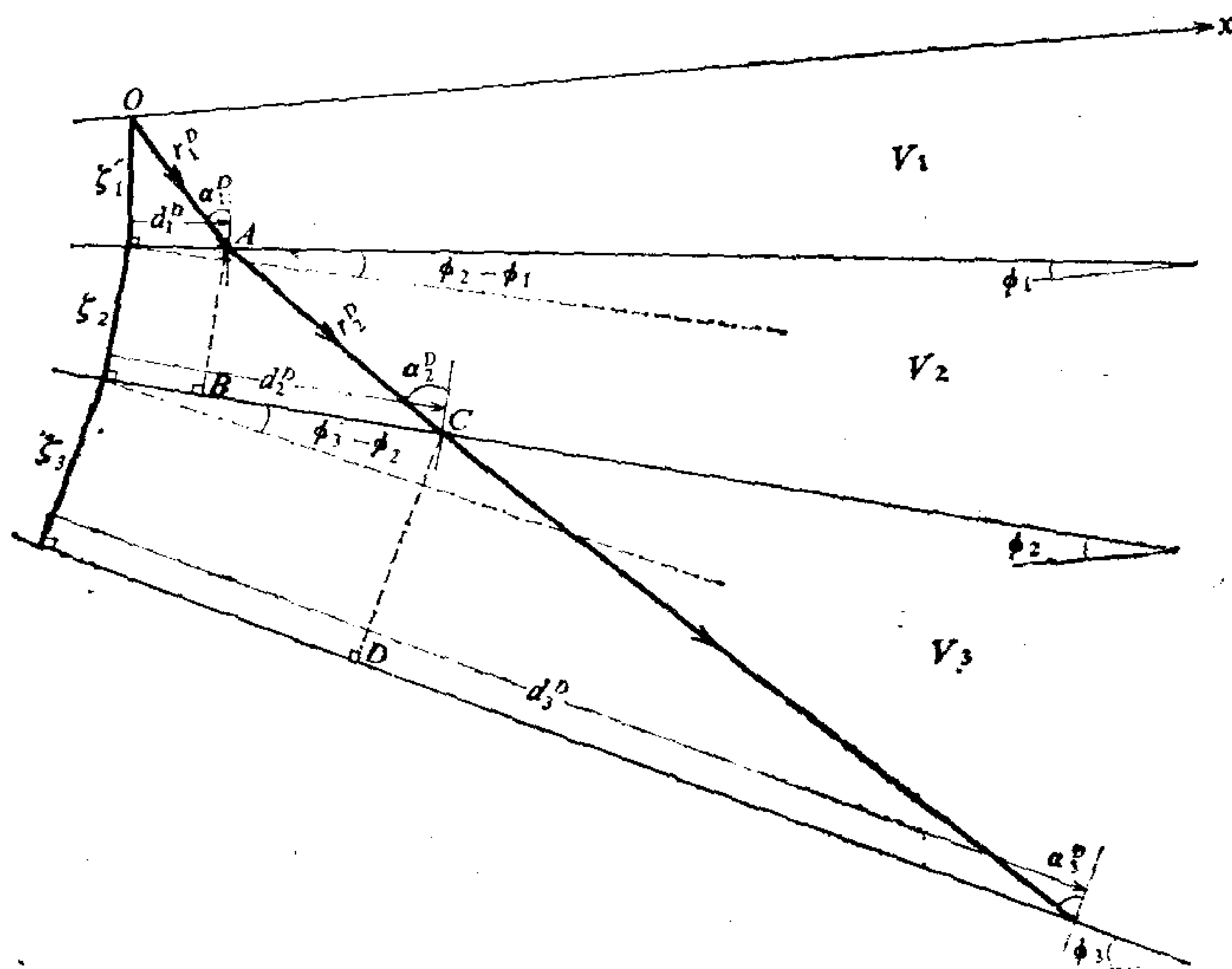



图 6-4-2

$$r_i^D = [\zeta_i + d_{i-1}^D \sin(\phi_i - \phi_{i-1})]/\cos\alpha_i^D, \qquad (6\text{-}4.3)$$

$$d_i^D = d_{i-1}^D \cos(\phi_i - \phi_{i-1}) + r_i^D \sin\alpha_i^D, \qquad (6\text{-}4.3')$$

$$i = 1, 2\cdots, n.$$

若地面坐标原点与炮点重合，恒有$d_0^D \equiv 0$及$\phi_0 \equiv 0$（地面水平线）。

对于上行波，如图6-4-3示，其递推公式为

$$r_i^U = [\zeta_i \cos(\phi_i - \phi_{i-1}) + d_i^U \sin(\phi_i - \phi_{i-1})]/\cos\beta_{i-1}^U, \qquad (6\text{-}4.4)$$

$$d_{i-1}^U = (d_i^U + r_i^U \sin\alpha_i^U)/\cos(\phi_i - \phi_{i-1}), \qquad (6\text{-}4.4')$$

$i=n, n-1, \cdots, 1$.

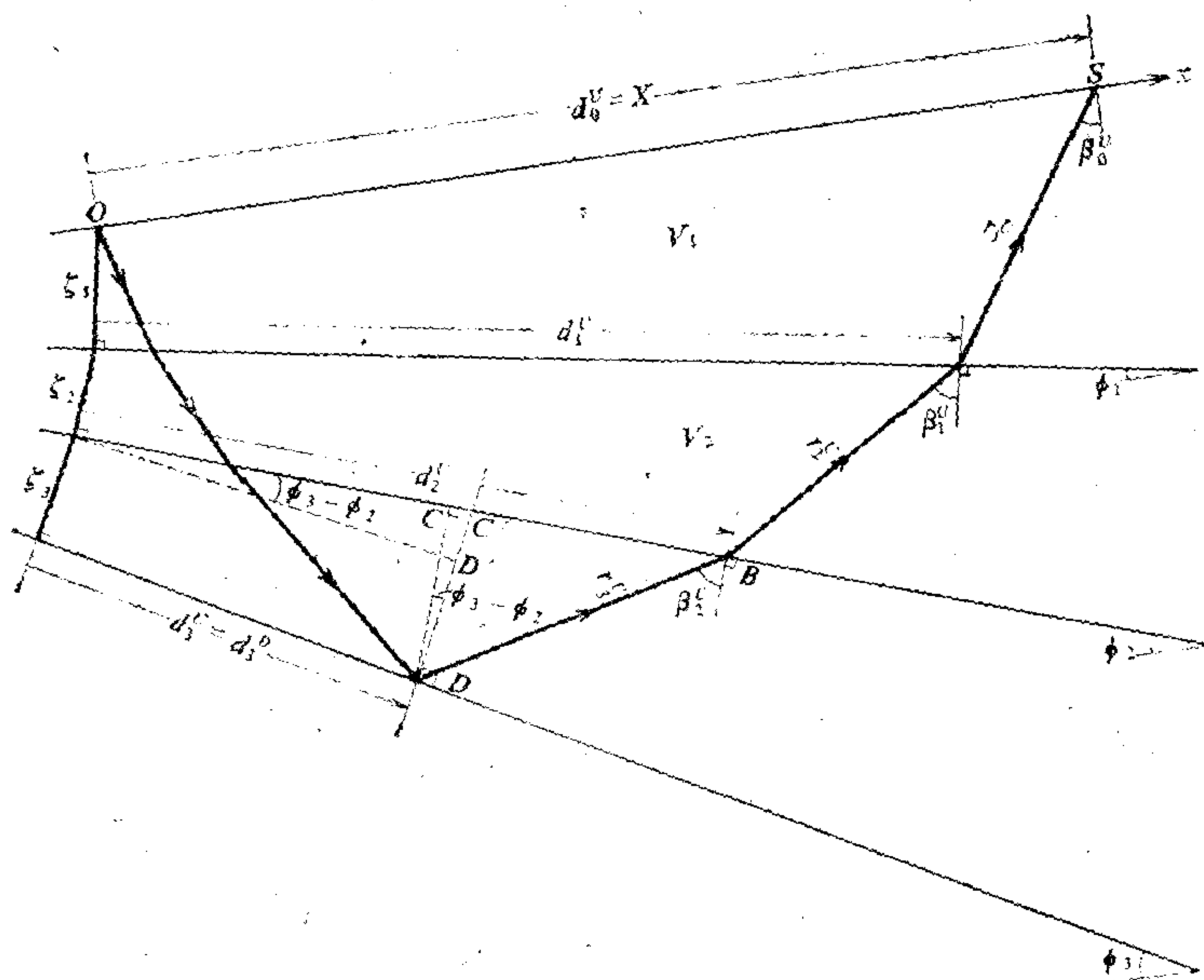



图 6-4-3

由(6-4.3)及(6-4.4)式求出各层反射波射线路径后，就可得到反射波通过倾斜层状介质的走时为

$$t=\sum_{i=1}^{n}\frac{r_i^D+r_i^U}{V_i}. \tag{6-4.5}$$

震中到接收点的距离X可对(6-4.4′)式令$i=0$求得

$$X=d_0^U. \tag{6-4.6}$$

计算步骤如下：

（1）给定各界面的倾角$\phi_i$及各层速度$V_i$，给定各层在原点处的垂直厚度$Z_i(i=1, 2, \cdots, n)$。

（2）对于一个震中距$X_j(j=1, 2, \cdots, m)$，用非线性求根法

找到对应$X_j$射线的初始入射角$(\beta_0^D)_j$。用$(\beta_0^D)_j$代入(6-4.1)式，递推求得该下行射线在各界面的入射角出射角$(\alpha_i^D)_j$及$(\beta_i^D)_j$($i=1$，2，…，$n$)；以及上行射线在各界面的出射角、入射角$(\beta_i^U)_j$及$(\alpha_i^U)_j$($i=n$，$n-1$，…，1)。

(3) 根据各层垂直厚度$Z_i$及界面倾角$\phi_i$，代入(6-4.2)及(6-4.2′)式，求出各层的层间垂直厚度$\zeta_i$ ($i=1$，2，…，$n$)。

(4) 将$\zeta_i$，$(\alpha_i^D)_j$及$\phi_i$代入(6-4.3)及(6-4.3′)递推求出各层的下行射线路径$(r_i^D)_j$，($i=1$，2，…，$n$)。将$\zeta_i$，$(\beta_i^U)$及$\phi_i$代入(6-4.4)及(6-4.4′)式，递推求出各层的上行射线路径$(r_i^U)_j$ ($i=n$，$n-1$，…，1)。

(5) 求相应于震中距$X_j$的理论走时(按(6-4.5))式有

$$t_j=\sum_{i=1}^{n}\frac{(r_i^D)_j+(r_i^U)_j}{V_i}.$$

下面讨论折射波(首波)走时的计算方法。如图 6-4-4 所示，对于倾斜界面常速层层状介质，地上第$K$个界面(自地下界面起算)产生的折射波的走时计算公式如下：

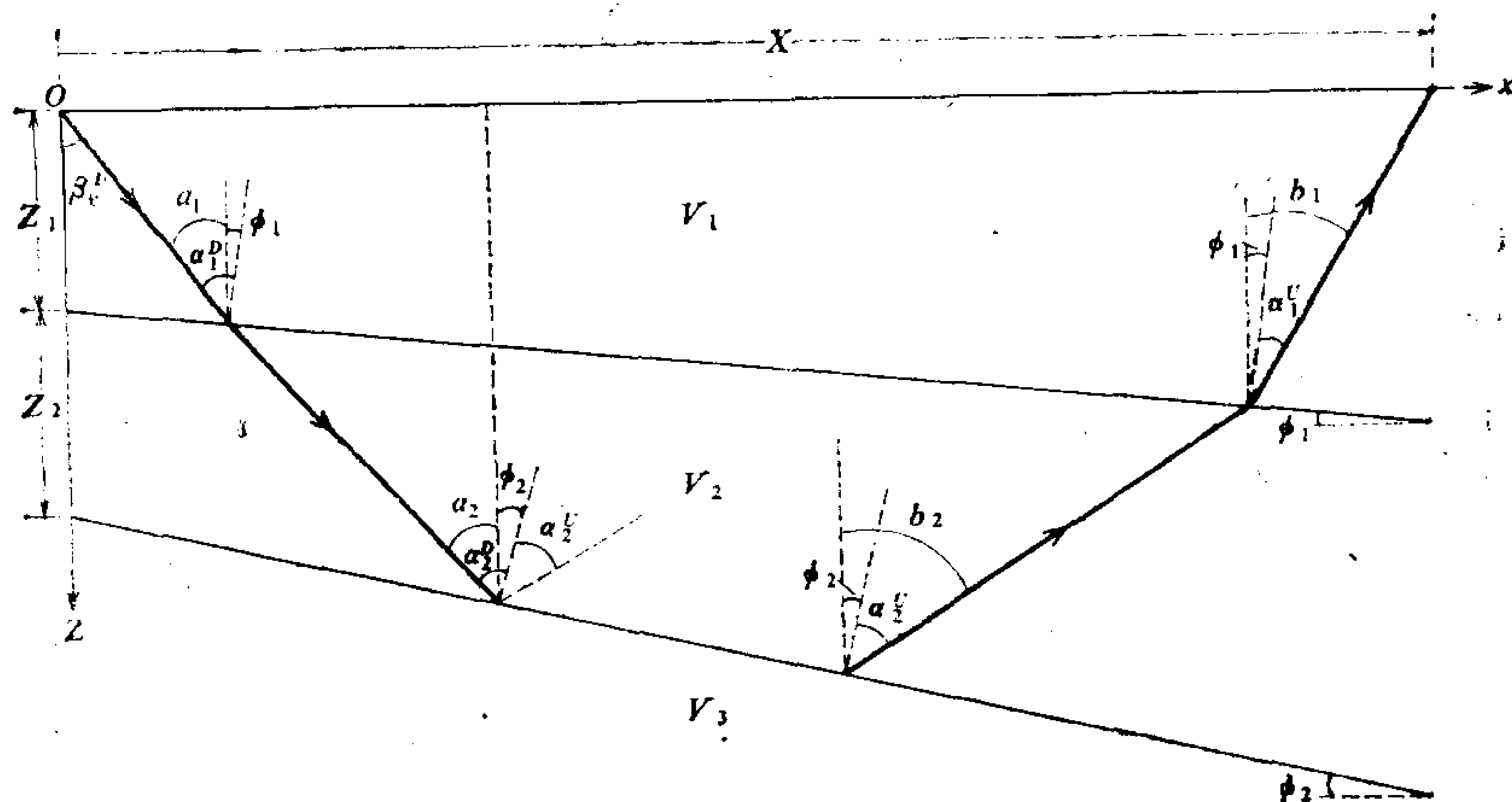



图6-4-4

$$t(K)=\frac{X\sin b_1}{V_1}+\sum_{i=1}^{K}\frac{Z_i}{V_i}(\cos a_i+\cos b_i). \qquad (6\text{-}4.7)$$

式中$K$表示产生折射波的界面的编号($K=1$，2，…，$n$)；

$Z_i$为震源$O$点地下第$i$层介质的垂直厚度；

$V_i$为第$i$层介质的波速；

$X$为震源$O$点到接收点的距离；

$a_i$为第$i$层介质中下行射线与垂线(平行于$Z$轴)的夹角。

$b_i$为第$i$层介质中上行射线与垂线之夹角。

由上式不难看出，要计算走时曲线，关键在于求得射线通过各界面的$a_i$及$b_i$角，这可由下述递推方法计算。由图6-4-4，不难得出下述关系：

$$\begin{aligned} a_i &= \alpha_i^D-\phi_i, \\ b_i &= \alpha_i^U+\phi_i. \end{aligned} \qquad (6\text{-}4.8)$$

式中$\alpha_i^D$及$\alpha_i^U$为下行或上行射线在$i$界面的入射角；$\phi_i$为界面倾角。

首先给定的层参数$Z_i$、$V_i$及$\phi_i$。对于第$K$个界面，根据全反射的临界角

$$\alpha_K^D=\sin^{-1}\left(\frac{V_K}{V_{K+1}}\right).$$

由折射波产生的条件，知出射角

$$\alpha_K^U\equiv\alpha_K^D.$$

利用(6-4.1)式，将$\alpha_K^U$及$\alpha_K^D$代入进行反复迭代，即可求出各界面的$\alpha_i^U$及$\alpha_i^D$值($i=K$，$K-1$，…，1)及反射线的入射角$\beta_0^D$，震中距$X$等。

由(6-4.8)式，可算出各界面的$a_i$及$b_i$角代入(6-4.7)式求得不同震中距$X_i$的到时$t_i(K)$。实际计算时，是先求出折射波出现的最小震中距$X_0$，当所取的$X_i>X_0$时，即计算折射走时，否则不进行计算。

## §6-5 用反射波走时反演地壳结构

本节讨论用地震反射波反演地壳结构的计算方法。反演计算的详细理论已在第一至第三章中谈到，下面仅简单说明其基本程序。

若地面沿观测剖面有$m$个观测点，各观测点走时向量为

$$\boldsymbol{t}^{观}=(t_1^{观},\ t_2^{观},\ \cdots,\ t_m^{观})^T.$$

对地壳结构给定一初始模型，这个模型用$n$个待反演的分层参数$\boldsymbol{x}=(x_1,\ x_2,\ \cdots,\ x_n)^T$。表示。则可算出相应于各测点的理论走时：

$$t^{理}(\boldsymbol{x})=(t_1^{理}(\boldsymbol{x}),t_2^{理}(\boldsymbol{x}),\cdots,\ t_m^{理}(\boldsymbol{x}))^T.$$

令走时残差为

$$\Delta\boldsymbol{t}=\boldsymbol{t}^{观}-\boldsymbol{t}^{理}(\boldsymbol{x}). \tag{6-5.1}$$

根据最小二乘原理，令下列目标函数取极小值

$$\Phi(\boldsymbol{x})=\|\Delta\boldsymbol{t}\|^2=\min, \tag{6-5.2}$$

可导致以下解答

$$\boldsymbol{A}^T\boldsymbol{A}\Delta\boldsymbol{x}=\boldsymbol{A}^T\Delta\boldsymbol{t}. \tag{6-5.3}$$

其中$\boldsymbol{A}$是$m\times n$阶微分系数矩阵（Jacobi矩阵），矩阵中各元素为

$$a_{ji}=\left.\frac{\partial t_j^{理}(\boldsymbol{x})}{\partial \boldsymbol{x}_i}\right|_{\boldsymbol{x}=\boldsymbol{x}_0},\qquad \begin{array}{l} j=1,\ 2,\ \cdots,\ m,\\ i=1,\ 2,\ \cdots,\ n. \end{array} \tag{6-5.4}$$

$\Delta\boldsymbol{x}$为层参数(厚度、速度等)的修正量

$$\Delta\boldsymbol{x}=\boldsymbol{x}-\boldsymbol{x}_0, \tag{6-5.5}$$

$\boldsymbol{x}_0$是参数的初值。$\Delta\boldsymbol{t}_0$系观测到时与由初值模型$\boldsymbol{x}_0$计算的理论到时的时差。利用阻尼最小二乘法可得式(6-5.3)的解为

$$\Delta\boldsymbol{x}=(\boldsymbol{A}^T\boldsymbol{A}+\lambda\boldsymbol{I})^{-1}\boldsymbol{A}^T\Delta t_0. \tag{6-5.6}$$

式中$\lambda$为阻尼系数，$\boldsymbol{I}$为单位矩阵。在解（6-5.6）式时，考虑到系

数矩阵的正定对称性，采用计算速度较高的Cholesky法计算。该方法可参考有关的计算方法参考书。

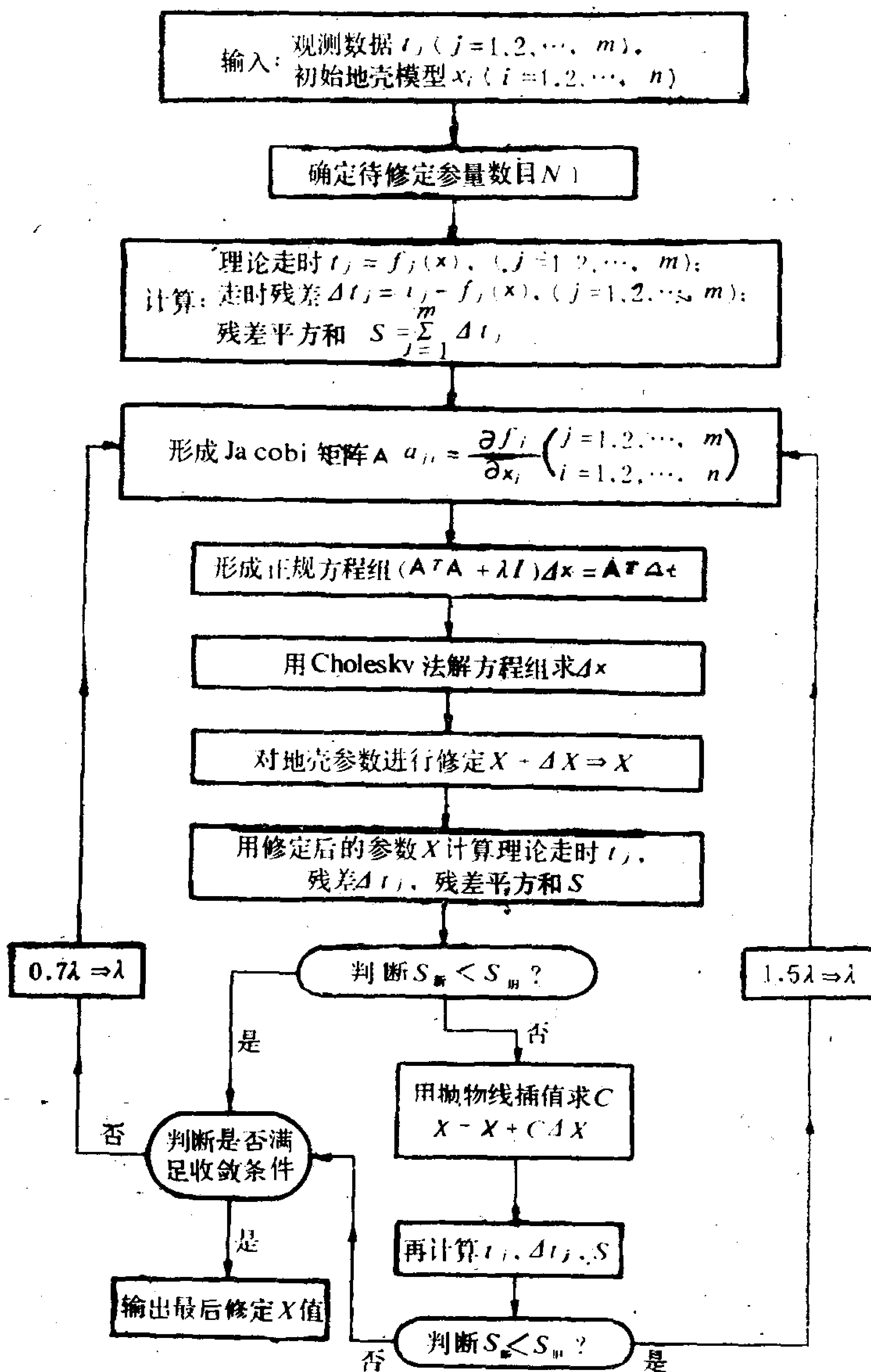
在修定参数的迭代过程中，可以用以下两个标准作为收敛条件，一是走时残差平方和达到预定的精度要求，二是模型参数改正量$\Delta x$已足够小，达到事先给定的限差。只要达到其中任一标准，则停止迭代、输出修定的地层参数。图 6-5-1 为用阻尼最小二乘法反演地壳结构的框图。

下面举一个数值反演实例，对于我国华北地区人工地震测深资料中一段记录，可以清楚地追踪到$T_2^s$、$T_3^s$、$T_4^s$三层反射波。采用逐层反演方法(即第一层反演后，将该层参数固定，然后反演第二层，以至第三层)。可得下列结果(均为折合走时$T-\Delta/6$)：

$T_2^s$ 反演数据

| 震中距$X$(km) | 理论计算走时$t^{理}$(s) | 观测走时$t^{观}$(s) |
|---|---|---|
| 16.32 | 6.36 | 6.18 |
| 18.54 | 6.58 | 6.66 |
| 20.23 | 6.75 | 6.87 |
| 28.43 | 7.74 | 7.78 |
| 37.68 | 9.02 | 9.10 |
| 43.18 | 9.85 | 9.85 |
| 50.30 | 10.95 | 10.93 |
| 56.38 | 11.93 | 11.80 |
| 61.48 | 12.76 | 12.81 |
| 63.76 | 13.13 | 13.23 |
| 66.01 | 13.51 | 13.52 |

残差平方和$S$=0.093s² 迭代次数：10次



输入：观测数据 $t_j$（$j=1,2,\cdots,m$），初始地壳模型 $x_i$（$i=1,2,\cdots,n$）
确定待修定参量数目 $N$
计算：理论走时 $t_j=f_j(x)$，（$j=1,2,\cdots,m$）；走时残差 $\Delta t_j=t_j-f_j(x)$，（$j=1,2,\cdots,m$）；残差平方和 $S=\sum_{j=1}^{m}\Delta t_j$
形成Jacobi矩阵A $a_{ji}=\dfrac{\partial f_j}{\partial x_i}\begin{pmatrix}j=1,2,\cdots,m\\i=1,2,\cdots,n\end{pmatrix}$
形成正规方程组 $(A^TA+\lambda I)\Delta x=A^T\Delta t$
用Cholesky法解方程组求 $\Delta x$
对地壳参数进行修定 $X+\Delta X\Rightarrow X$
用修定后的参数 $X$ 计算理论走时 $t_j$，残差 $\Delta t_j$，残差平方和 $S$
判断 $S_{新}<S_{旧}$？
$0.7\lambda\Rightarrow\lambda$
$1.5\lambda\Rightarrow\lambda$
是
否
用抛物线插值求 $C$ $X=X+C\Delta X$
判断是否满足收敛条件
再计算 $t_j,\Delta t_j,S$
是
输出最后修定 $X$ 值
判断 $S_{新}<S_{旧}$？
否
是


图6-5-1

**$T_3^0$ 反演数据**

| 震中距$X$ | 理论计算走时$t^{理}$ | 观测走时$t^{观}$ |
|---|---|---|
| 28.43 | 9.50 | 9.60 |
| 37.68 | 10.48 | 10.54 |
| 40.15 | 10.77 | 10.78 |
| 43.18 | 11.14 | 11.10 |
| 47.35 | 11.67 | 11.62 |
| 51.79 | 12.25 | 12.18 |
| 56.38 | 12.87 | 12.88 |
| 63.76 | 13.91 | 14.06 |
| 66.01 | 14.24 | 13.30 |

残差平方和$S$=0.077$s^2$　迭代次数12次

**$T_4^0$ 反演数据**

| 震中距 | 理论计算走时$t^{理}$ | 观测走时$t^{观}$ |
|---|---|---|
| 51.79 | 14.62 | 14.66 |
| 56.38 | 14.99 | 14.93 |
| 61.48 | 15.43 | 15.49 |
| 63.76 | 15.63 | 15.62 |

残差平方和$S$=0.009$s^2$　迭代次数：25次

各层参数逐次反演结果

| 层号 | 初值 V(km/s) | 初值 h (km) | 反演计算值 V(km/s) | 反演计算值 h (km) |
|---|---|---|---|---|
| 第一层 | 4.00 | 3.14 | | |
| 第二层 | 5.00 | 10.00 | 5.65 | 11.25 |
| 第三层 | 7.00 | 10.00 | 6.15 | 7.53 |
| 第四层 | 7.00 | 15.00 | 8.16 | 17.85 |

注：第一层速度及厚度选定不变

以上反演结果表明，该区地壳大体可分为沉积层、地壳上层、地壳中层和地壳下层几部分。其层速度一般是递增的。

## §6-6 地壳模型及震源参数的同时反演

前面，我们讨论了地面震源层状介质地震体波走时的正演及反演问题。由于震源位置及发震时刻均是已知的，所以待求的仅是各地壳分层参数，对于天然地震源来说，由于震源参数及地壳模型两者都是未知的，因此其反演问题要复杂一些。

1976年，Crosson研究了用地震台网记录的地方震与近震的观测资料同时反演地壳速度结构及震源参数的方法。这个模型仅限于地壳为水平常速分层，地震可以发生在地下任意深度。这时，$P$波的初至可能出现两种情况：直达波$\overline{P}$或折射波$P_n$如图6-6-1所示。

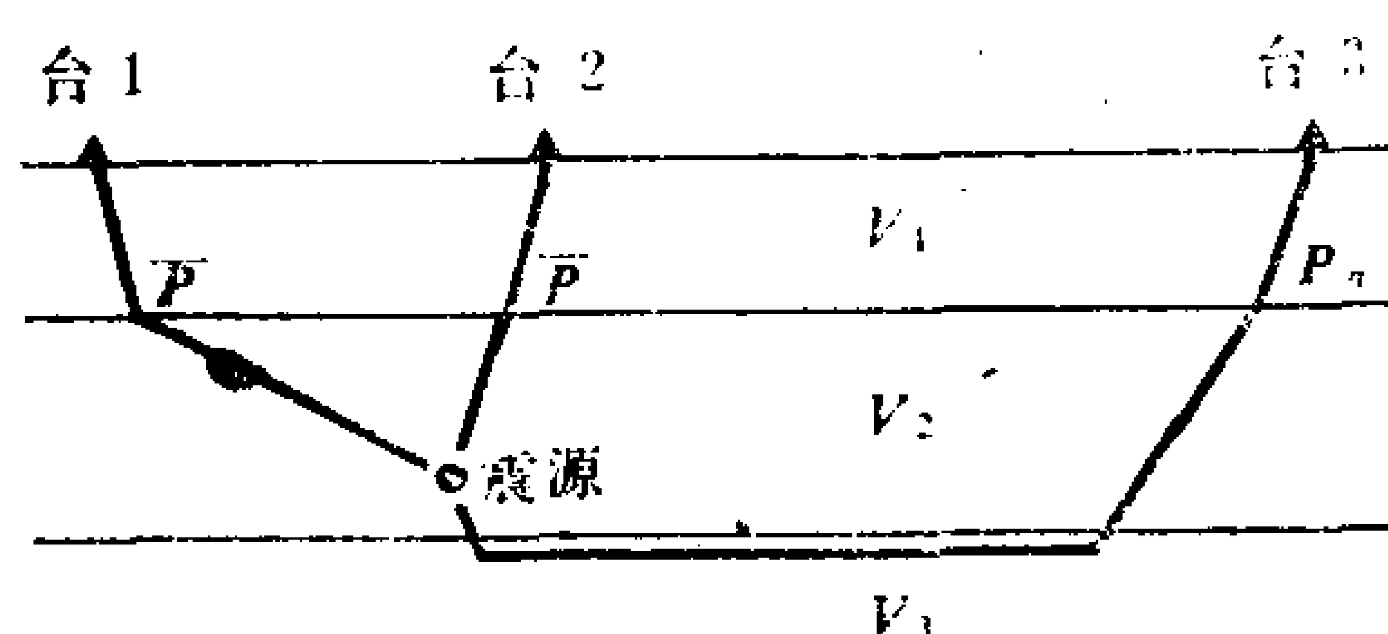



图 6-6-1

### （一）正演问题

下面讨论一系列常速水平层所组成的地壳模型，任意深度震

源的直达波及折射波走时的计算问题。

我们先考虑直达波。若震源位于第一层中，那末直达波是简单的、可以用简单的解析式$t=\sqrt{\Delta^2+Z^2}/v_1$计算。然而，若震源位于第二或更深的层内，就不再有可利用的解析式，这时，必须利用迭代法寻求与相应震中距的离源角$\phi$、如图 6-6-2所示。

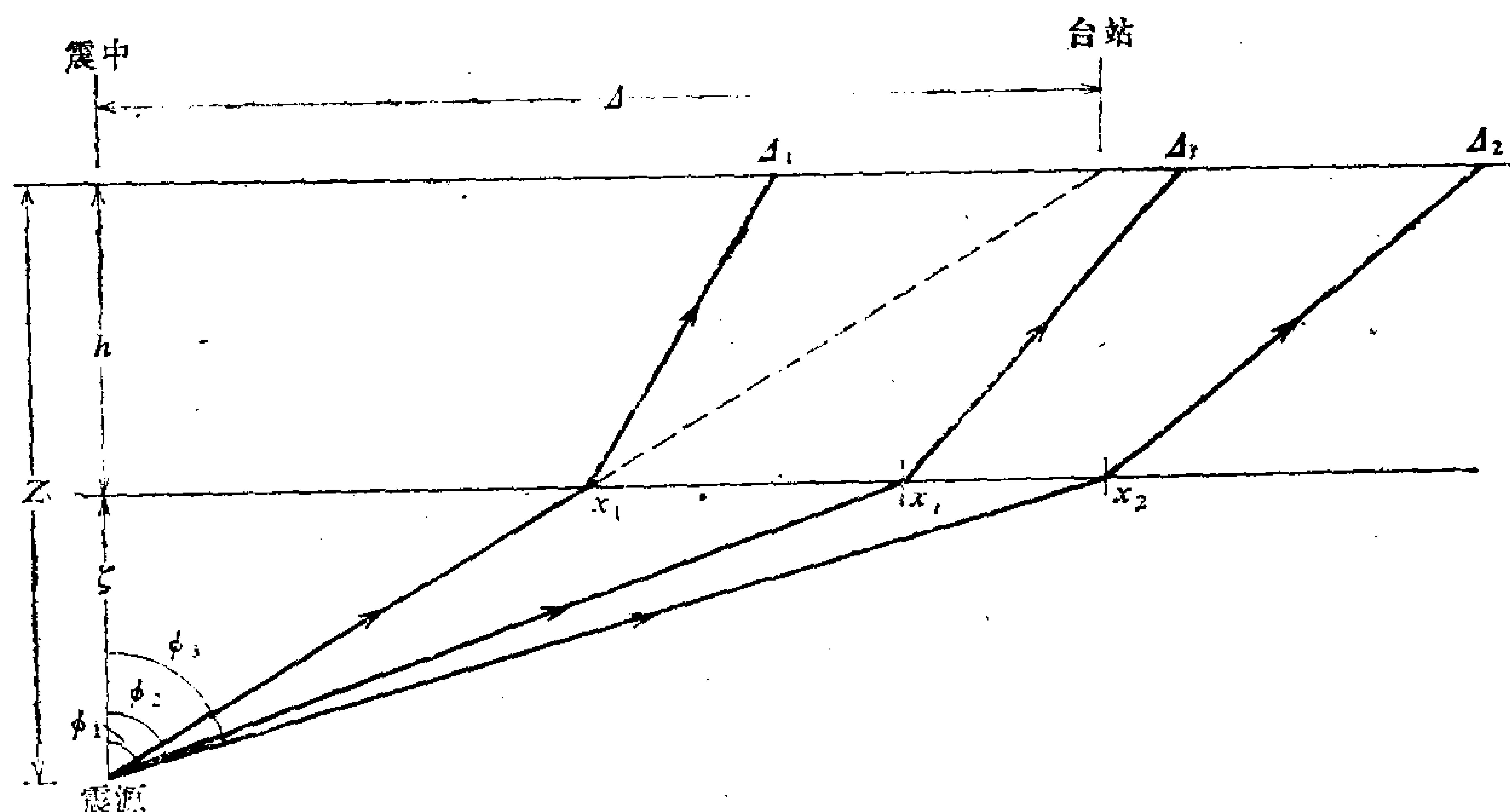



图 6-6-2

设震源在第二层内，并设已知台站的震中距为$\Delta$。首先选择角$\phi_1$及$\phi_2$，使$\phi_1<\phi<\phi_2$。一般可选择$\phi_1$使得$x_1=\Delta\frac{\zeta}{Z}$，选择$\phi_2$使$x_2=\Delta$。根据离源角为$\phi_1$及$\phi_2$两条出射线，能方便地算出地面相应的震中距$\Delta_1$及$\Delta_2$。

按下式计算试验值$x_t$：

$$\frac{x_t-x_1}{x_2-x_1}=\frac{\Delta-\Delta_1}{\Delta_2-\Delta_1}. \tag{6-6.1}$$

然后，用$x_t$(相应的离源角为$\phi$)计算震中距$\Delta_t$。检验$|\Delta_t-\Delta|$是否达到给定限差(即小于误差要求)。如未达到，则用第一次迭代求得的试验值$\Delta_t$作为一个新值。若$\Delta_t>\Delta$，取$x_1$和$x_t$作为新的搜寻范围；若$\Delta_t<\Delta$，则用$x_t$和$x_2$作为新的搜寻范围。重复同样过程，直到最后求出的$\Delta_t$满足$|\Delta_t-\Delta|\leqslant\varepsilon$ ($\varepsilon$为给定限差)为止。

上述迭代过程容易推广到震源在更深层的情况。一旦给出了第$n$层离源角$\phi=\theta_n$，我们就可以用Snell定律计算上覆各层的出射角，

$$\frac{\mathrm{Sin}\theta_n}{v_n}=\frac{\mathrm{Sin}\theta_{n-1}}{v_{n-1}}=\cdots\cdots=\frac{\mathrm{Sin}\theta_1}{v_1}=p. \qquad (6\text{-}6.2)$$

或者写为

$$\mathrm{Sin}\theta_i=pv_i, \qquad (i=n,\ n-1,\ \cdots,\ 1). \qquad (6\text{-}6.3)$$

迭代过程所计算的震中距分别可以表示为

$$\Delta_1=x_1+\sum_{i=n-1}^{1}h_i\tan\theta_i, \qquad \text{当 } \theta_n=\phi_1\text{时；}$$

$$\Delta_2=x_2+\sum_{i=n-1}^{1}h_i\tan\theta_i, \qquad \text{当 } \theta_n=\varphi_2\text{时。} \qquad (6\text{-}6.4)$$

按(6-6.1)式计算$x_l$，重复此步骤，直到$|\Delta_l-\Delta|\leqslant\varepsilon$为止。震中距及走时的计算公式如下：

$$\Delta_l=x_l+\sum_{i=n-1}^{1}h_i\tan\theta_i,$$

$$t=\frac{\sqrt{x_l^2+\zeta^2}}{v_n}+\sum_{i=n-1}^{1}\frac{h_i}{v_i\cos\theta_i}, \qquad (6\text{-}6.5)$$

当

$$\theta_n=\sin^{-1}\left(\frac{x_l}{\sqrt{x_l^2+\zeta^2}}\right).$$

下面讨论折射波走时的计算方法。对于$n$个水平层的情况，如图6-6-3所示。折射波走时可以用解析公式一次求得

$$t_k=\frac{\Delta}{v_k}+2\sum_{i=1}^{k-1}\frac{h_i}{v_i}\cos\theta_{ik},\ k=2,\ 3,\ \cdots,\ n. \qquad (6\text{-}6.6)$$

$\theta_{ik}$之间的关系由Snell定律给出，一般可写为

$$\mathrm{Sin}\theta_{ik}=\frac{v_i}{v_k}, \qquad i=1,\ 2,\ \cdots,\ n-1;\quad k=2,\ 3,\ \cdots,\ n. \qquad (6\text{-}6.7)$$

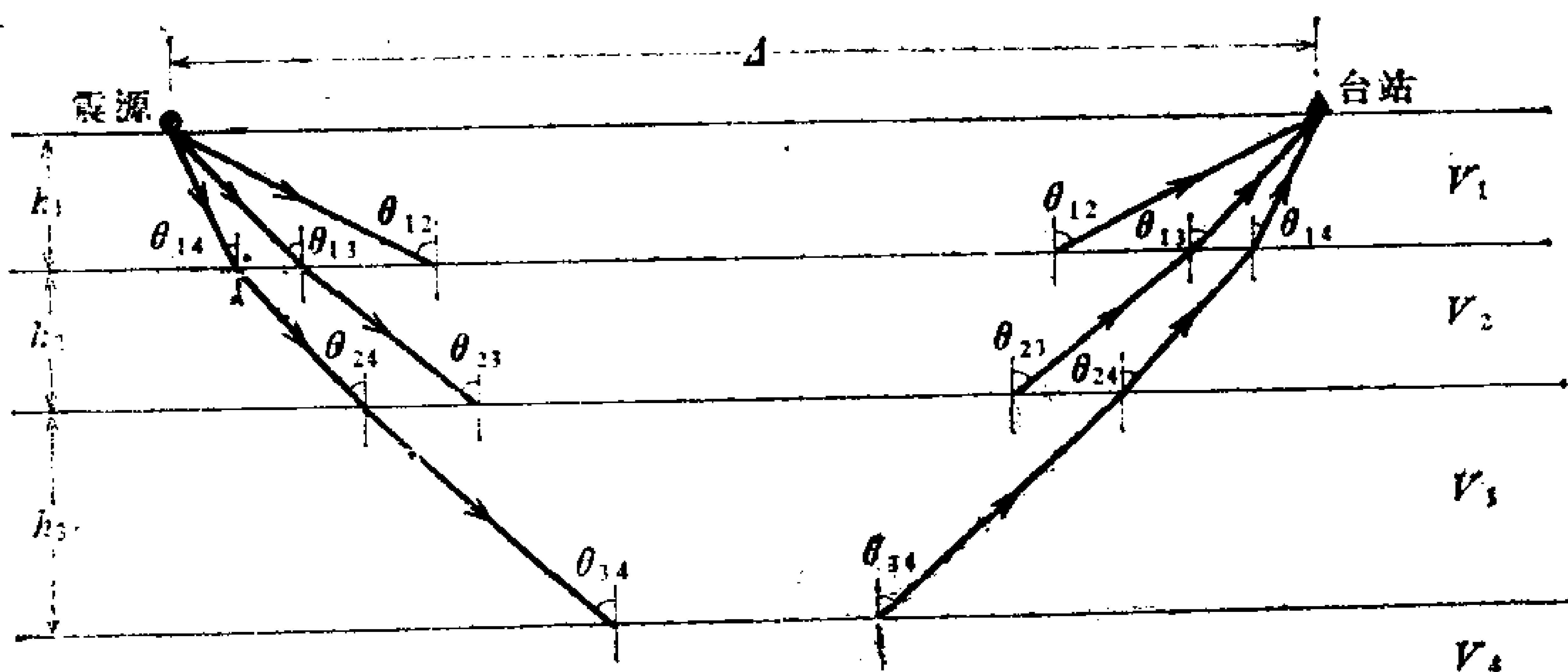



图 6-6-3

利用关系 $\cos^2\theta_{ik}=1-\sin^2\theta_{ik}$，则(6-6.6)式可以直接用层速度参数表达：

$$t_k=\frac{\Delta}{v_k}+2\sum_{i=1}^{k-1}\frac{h_i\sqrt{v_k^2-v_i^2}}{v_iv_k}\text{，}k=2\text{，}3\text{，}\cdots\text{，}n\text{。}\tag{6-6.8}$$

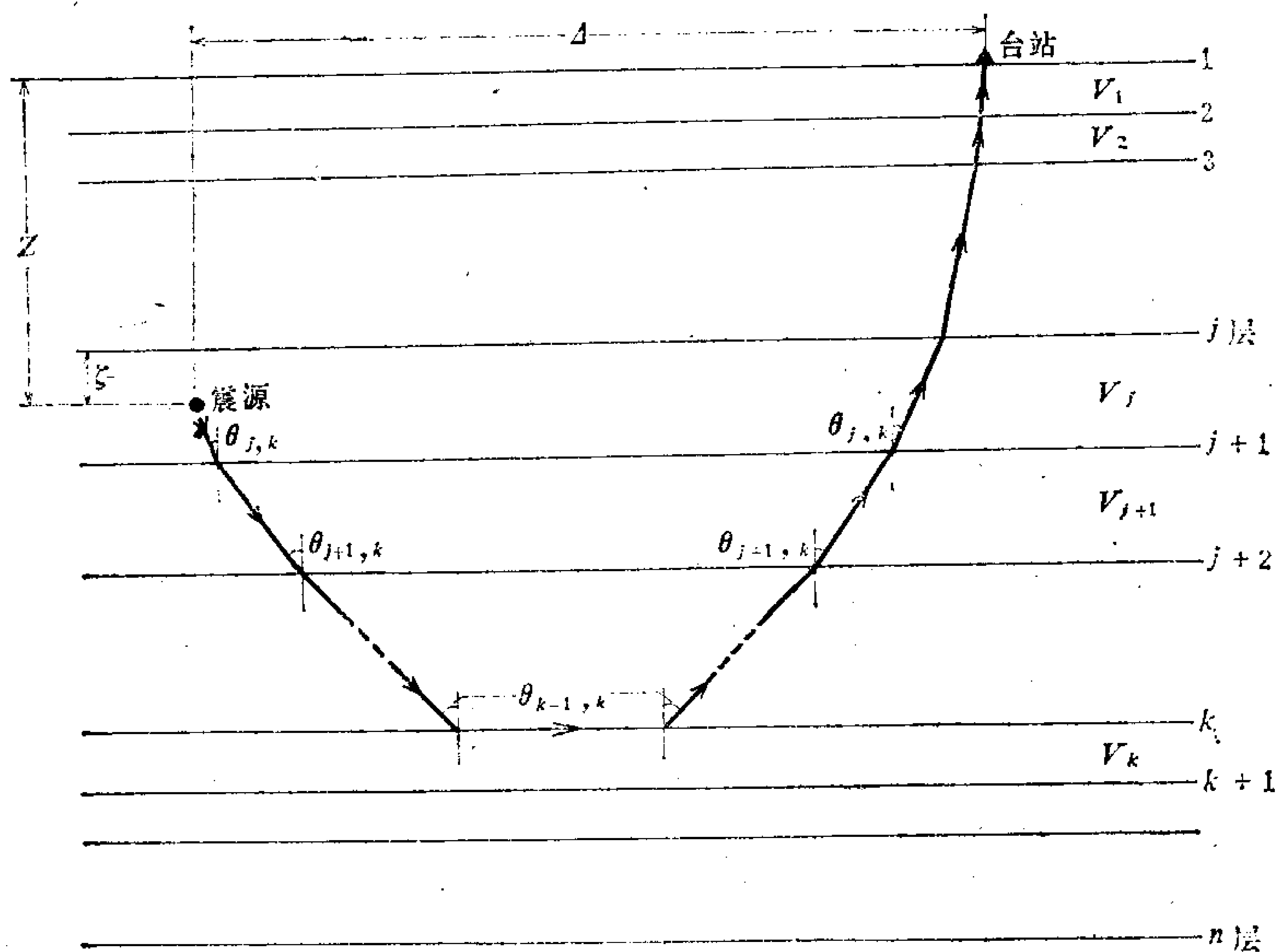



图 6-6-4

现在讨论震源位于任意深度Z上的一般情况。设震源位于第$k$层之内。震源至第$k$层顶部距离为$\zeta$，如图6-6-4所示。

这种情况下与表面震源的区别在于，折射波的下行走时路径从深度Z处开始而不是从表面开始。这意味着下行的震波少走$j$层（加上厚度为$\zeta$速度为$v_i$的层）。因此可以写出类似于(6-6.8)式的方程，

$$t_{jk}=\frac{\Delta}{v_k}-\frac{\zeta\sqrt{v_k^2-v_j^2}}{v_j v_k}+\sum_{i=1}^{j-1}\frac{h_i\sqrt{v_k^2-v_i^2}}{v_i v_k}+2\sum_{i=j}^{k-1}\frac{h_i\sqrt{v_k^2-u_i^2}}{v_i v_k}\ ; \qquad (6\text{-}6.9)$$

$j=1,\ 2,\ \cdots,\ n-1;\quad k=2,\ 3,\ \cdots,\ n.$

这一公式可以计算任意埋深震源的折射波的到时。

对于某一台站而言，要得到其$P$波最小走时，可以先计算折射波出现的临界距离

$$Z_{jk}=\sum_{i=1}^{j=1}\frac{h_i v_i}{\sqrt{v_k^2-v_i^2}}+2\sum_{i=j}^{k-1}\frac{h_i v_i}{\sqrt{v_k^2-v_i^2}}-\frac{\zeta v_i}{\sqrt{v_k^2-v_i^2}}\ ;$$

$j=1,\ 2,\ \cdots,\ n-1;\quad k=2,\ 3,\ \cdots,\ n.\qquad (6\text{-}6.10)$

若台站震中距$\Delta<X_{jk}$，则直达波先到达，计算直达波走时作为$P$波到台站的最小走时，反之若$\Delta>X_{jk}$，则折射波先到达，计算折射波走时为$P$波最小走时。

## （二）反演问题

尽管地震$P$波走时一般是震源位置和速度模型参数的非线性函数，但可以用Taylor一阶展开式将其表示为拟线性方程组。设在$r$个台站上记录到$q$次地震，即总的地震到时有$r\times q$个记录。

第$i$次地震在第$j$个台站上的$P$波最小到时为

$$t_{ij}=t_{ij}(x_{1i},\ x_{2i},\ x_{2i},\ x_{4i},\ a_1,\ a_2,\ \cdots,\ a_e);$$

$i=1,\ 2,\ \cdots,\ q;\quad j=1,\ 2,\ \cdots,\ r.\qquad (6\text{-}6.11)$

式中$x_{1i}$，$x_{2i}$，$x_{3i}$及$x_{4i}$为第$i$次地震震源参数，$\alpha_1, \alpha_2, \cdots, \alpha_e$为地壳分层的厚度及速度等参数。由震源和模型参数的微小变化引起的走时变化可用下式计算：

$$\Delta t_{ij} = \sum_{k=1}^{4}\left(\frac{\partial t_{ij}}{\partial x_{ki}}\right)\Delta x_{ki} + \sum_{k=1}^{e}\left(\frac{\partial t_{ij}}{\partial \alpha_k}\right)\Delta \alpha_k ;$$

$$i=1,\ 2,\ \cdots,\ q;\ j=1,\ 2,\ \cdots,\ r. \qquad (6\text{-}6.12)$$

式中$\Delta t_{ij} = t_{ij} - t^{0}_{ij}$，$\Delta x_{ki} = x_{ki} - x^{0}_{ki}$，$\Delta \alpha_k = \alpha_k - \alpha^{0}_k$；$x^{0}_{ki}$及$\alpha^{0}_k$分别为震源及地壳模型参数的初值。在作反演计算时，通常将$\Delta t_{ij}$看成观测值与计算值之差。

基于震源及地壳模型的初值，可用解析法或数值法计算出(6-6.12)式中的各项偏微分。式 6-6.12)展开后可以写为$m=qr$个方程，$n=4q+l$个未知数的线性方程组。记为

$$\mathbf{A}\Delta \boldsymbol{x} = \Delta \boldsymbol{t}. \qquad (6\text{-}6.13)$$

式中$\boldsymbol{A}$为$m\times n$系数矩阵，$\Delta\boldsymbol{x}$为$n$维向量，$\Delta\boldsymbol{t}$中$m$维向量，该矩阵具有下列形式：

$$\begin{pmatrix}
\frac{\partial t_{11}}{\partial x_{11}} & \cdots & \frac{\partial t_{11}}{\partial x_{41}} & & & & \frac{\partial t_{11}}{\partial \alpha_1} & \cdots & \frac{\partial t_{11}}{\partial \alpha_l} \\
\vdots & & \vdots & & 0 & & \vdots & & \vdots \\
\frac{\partial t_{1r}}{\partial x_{11}} & \cdots & \frac{\partial t_{1r}}{\partial x_{41}} & & & & \frac{\partial t_{1r}}{\partial \alpha_1} & \cdots & \frac{\partial t_{1r}}{\partial \alpha_l} \\
 & & & \frac{\partial t_{21}}{\partial x_{12}} & \cdots & \frac{\partial t_{21}}{\partial x_{42}} & & & \\
 & 0 & & \vdots & & \vdots & \vdots & & \vdots \\
 & & & \frac{\partial t_{2r}}{\partial x_{12}} & \cdots & \frac{\partial t_{2r}}{\partial x_{42}} & & & \\
 & \vdots & & & & \ddots & & & \\
 & 0 & & & 0 & & \frac{\partial t_{qr}}{\partial \alpha_1} & \cdots & \frac{\partial t_{qr}}{\partial \alpha_l}
\end{pmatrix}
\begin{pmatrix}
\Delta x_{11} \\ \vdots \\ \Delta x_{41} \\ \Delta x_{12} \\ \vdots \\ \Delta x_{42} \\ \vdots \\ \Delta x_{1q} \\ \vdots \\ \Delta x_{4q} \\ \Delta \alpha_1 \\ \vdots \\ \Delta \alpha_l
\end{pmatrix}
=
\begin{pmatrix}
\Delta t_{11} \\ \vdots \\ \Delta t_{1r} \\ \Delta t_{21} \\ \vdots \\ \Delta t_{2r} \\ \vdots \\ \Delta t_{q1} \\ \vdots \\ \Delta t_{qr}
\end{pmatrix}$$

(6-6.14)

首先对震源参数及地壳模型初值赋值，则系数矩阵A及走时残差$\Delta t$就可以确定。用广义逆方法或阻尼最小二乘法解(6-6.13)式，求得参数的校正量$\Delta x$。用迭代法对参数逐次修定，直到走时残差或各参数修改量达到预定精度为止。

式(6-6.13)中待定的参数量是很大的。例如，对于40次地震，地壳分为10层的情况，总共有$4\times40+2\times10=180$个未知数。仅贮存系数矩阵就要32400个单元。但考虑到正规方程组的系数矩阵$A^TA$是对称的，因此仅需存贮主对角线以上的系数值。

最后，我们将Crosson从美国华盛顿州西部Puget Sound地区的14个地震台记录中，挑选出40个最好的三级以下地方震记录(初至清晰，而且均匀分布于不同深度)，用$P$波初至到时反演震源参数及地壳模型。最后得到$A$、$B$两个模型。除地壳内低速带细节不同外，二者基本是相同的。对各次地震震源参数反演结果也较一致。如图6-6-5所示。

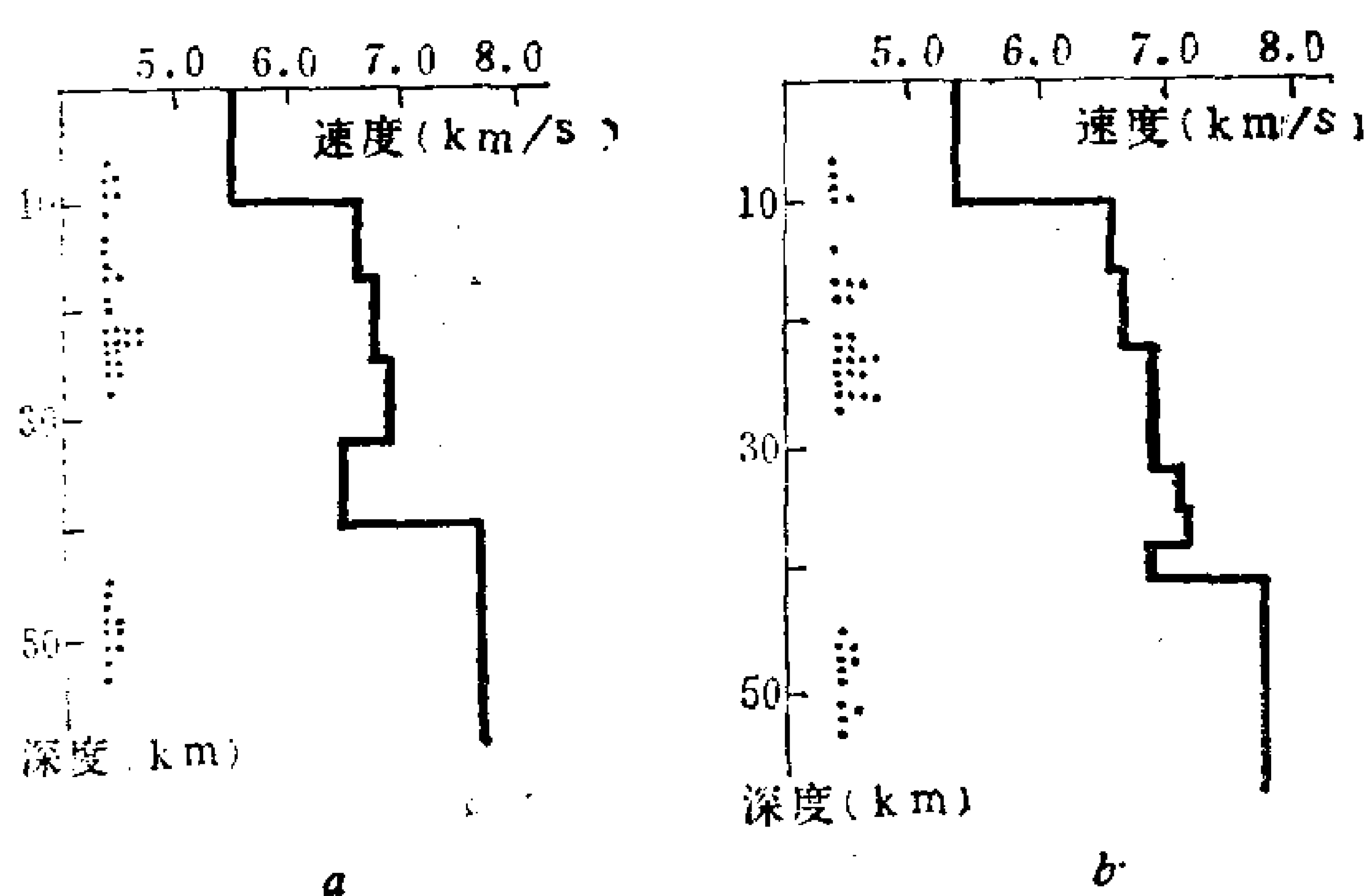



图 6-6-5

## §6-7 用剥去法反演地壳速度分布

本节讨论用回折波走时反演地壳速度结构的剥去法。它建立在球对称分层介质模型基础上，对于人工或天然地震震源的地壳上地幔探测都适用。下面分别讨论其正演及反演问题。

对于球层纵向速度变化规律设为

$$V=V_1\left(\frac{r}{r_1}\right)^{\lambda},\qquad (6\text{-}7.1)$$

式中$V_1$及$r_1$为球层顶部速度及距地心的半径。$\lambda$为球层速度按指数变化常数

$$\lambda=\frac{\ln\left(\frac{V_2}{V_1}\right)}{\ln\left(\frac{r_2}{r_1}\right)}. \qquad (6\text{-}7.2)$$

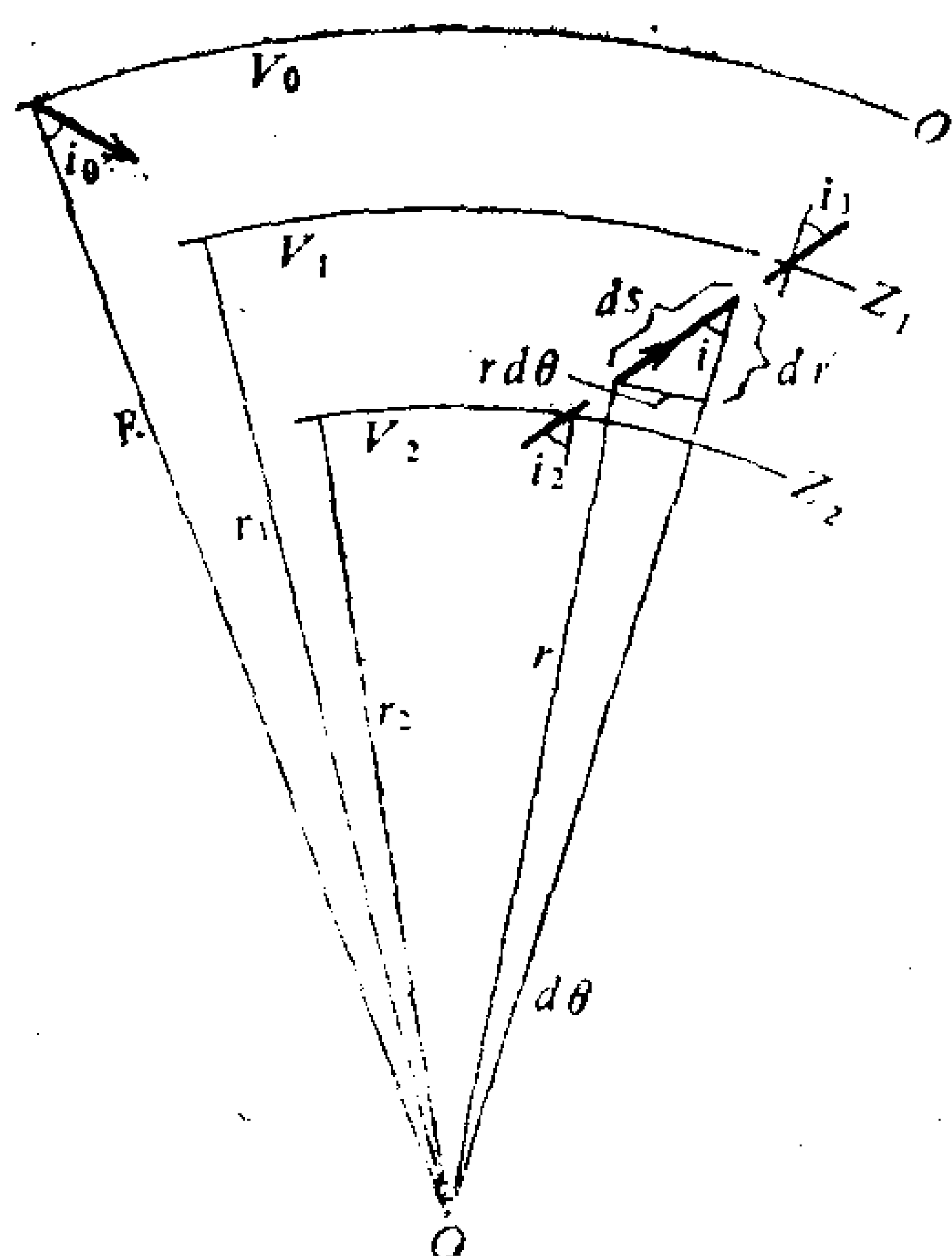


图 6-7-1

$V_2$与$r_2$分别为球层底部速度及距地心的半径。球层分布如图6-7-1所示。

由式（6-7.2）不难看出，$\lambda<0$表示正速度梯度层；$\lambda=0$为常速层；$\lambda>0$为负速度梯度层。

对于球层介质中的Snell定律为（参看§7-1）

$$p=\frac{r\sin i}{V},$$

或改写为

$$i=\mathrm{Sin}^{-1}\frac{pV}{r}, \qquad (6\text{-}7.3)$$

对（6-7.3）式微分并化简可得

$$di=(\lambda-1)\,\mathrm{tg}\,i\frac{dr}{\gamma}, \qquad (6\text{-}7.4)$$

或

$$di=\frac{(\lambda-1)\mathrm{Sin}^2 i}{p\cos i}\,\frac{dr}{V}, \qquad (6\text{-}7.5)$$

由图6-7-1可知有关系

$$ds=\frac{dr}{\cos i},$$

及

$$rd\theta=\mathrm{tg}\,i\cdot dr,$$

于是有 $$dt=\frac{ds}{V}=\frac{dr}{V\cos i},\tag{6-7.6}$$

及 $$d\theta=\operatorname{tg}i\frac{dr}{r},\tag{6-7.7}$$

将(6-7.4)式代入(6-7.7)，(6-7.5)式代入(6-7.6)，积分限从$i_2$至$i_1$，可得

$$\theta=\int_{i_2}^{i_1}d\theta=\frac{1}{\lambda-1}(i_1-i_2),\tag{6-7.8}$$

$$t=\int_{i_2}^{i_1}dt=\frac{p}{\lambda-1}(\operatorname{ctg}i_2-\operatorname{ctg}i_1),\tag{6-7.9}$$

注意，这里$\theta$及$t$仅代表射线通过某一球层的时间及距离的一半，故通过$N$层的总时间和距离可由以下求和公式计算

$$T=2\sum_{k=1}^{N}\frac{p}{\lambda_k-1}(\operatorname{ctg}i_{2k}-\operatorname{ctg}i_{1k})\tag{6-7.10}$$

$$\theta=2\sum_{k=1}^{N}\frac{1}{\lambda_k-1}(i_{1k}-i_{2k})\tag{6-7.11}$$

式中 $$i_{1k}=\sin^{-1}\frac{pV_{1k}}{r_{1k}},$$

$$i_{2k}=\sin^{-1}\frac{pV_{2k}}{r_{2k}},$$

$r_{1k}$、$r_{2k}$为第$k$层上、下界面半径，$V_{1k}$、$V_{2k}$为第$k$层上、下界面速度。射线参数$p$可由初始入射角$i_0$、地表介质速度$V_0$及地球半径$R$求得

$$p=\frac{R\sin i_0}{V_0}.$$

在用剥去法解反演问题时，是将地壳介质划分为许多薄层，这样可以假定每一层为常速层，即$\lambda=0$，于是(6-7.8)及(6-7.9)式写为

$$\theta = i_2 - i_1 \tag{6-7.12}$$

$$t = p(\mathrm{ctg} i_1 - \mathrm{ctg} i_2), \tag{6-7.13}$$

由于以上二式中$\theta$及$t$均可由观测走时曲线得到，对于地表附近一点 $\theta = \dfrac{\Delta_{观测}}{2R}$ ，$t = \dfrac{T_{观测}}{2}$ ，由第一层开始反演，由走时曲线求得地表附近回折波的视速度$V^*$，得出相应的$p$值，

$$p = \frac{R}{V_1/\sin i_1} = \frac{R}{V^*}$$

代入(6-7.12)及(6-7.13)式，可以解出$i_1$及$i_2$值：

$$\left.\begin{aligned} i_1 &= \frac{1}{2}\left[\cos^{-1}\left(\cos\theta - \frac{2p}{t}\sin\theta\right) - \theta\right], \\ i_2 &= \theta - i_1. \end{aligned}\right\} \tag{6-7.14}$$

设第一层顶界面$r_1 = R$，再根据式$r_2 = R\sin i_1/\sin i_2$，求出第一层的下界深度

$$Z_2 = R - r_2, \tag{6-7.15}$$

及该层速度

$$V_1 = R\sin i_1/p. \tag{6-7.16}$$

剥去第一层后，输入下一点的距离、走时及射线参数

$$\theta = \frac{\Delta}{2R},$$

$$t = \frac{T}{2},$$

及

$$p = \frac{R}{V^*}.$$

这时，应消去第一层走时及距离效应

$$\theta' = \theta - (i_2 - i_1),$$

$$t' = t - p(\mathrm{ctg} i_1 - \mathrm{ctg} i_2),$$

将$\theta'$及$t'$代入(6-7.14)式，即可计算第二层相应的$i_1$、$i_2$角，再代入(6-7.15)及(6-7.16)式，求出第二层相应的深度及速度。

如此递推，即可求得一系列深度及速度值。在用剥去法计算速度深度关系时，必须对走时曲线仔细加工，一般要用多项式或样条函数拟合走时曲线后，再根据它计算视速度。

## §6-8 层状介质地震波走时正、反演程序

### （一）层状介质地震波理论走时正演程序

本程序用于计算地表激发震源任意多层水平层状介质（常速层或速度梯度层）的反射波、回折波及折射波的理论走时。

**（一）计算方法**

本程序统一处理每一层或每一界面的反射波，回折波及首波的计算问题。在主程序中，首先输入层状介质模型的各种参数。然后对每一层的速度特征进行判断。如果是速度梯度层，则计算回折波及反射波、折射波等，否则只计算反射波及折射波。从上到下逐层计算，直到最下面一个界面为止。如果速度梯度层上（下）界面的速度与上（下）层的速度不同，可在该处假定一个厚度为零的常速层。本程序可以计算通过低速层的各类波的走时。

对于反射波给定的震中距，用弦位法求根计算相应的初始入射角。对于回折波是按入射角度的变化而计算走时的。对于首波则按给定等间隔距离计算走时。下面给出计算框图。（图6-8-1）

**（二）主要标识符及子程序说明**

THKNES(20)——水平层状介质各层厚度值（最多可计算20层）。

VP(20)——各层介质的波速。

AR(20)——各层的速度梯度值。

UV(20)——表示各层速度的特征参量

$$UV(I)=\begin{cases}1.0 & \text{速度梯度层，}\\ 0.0 & \text{常速层。}\end{cases}$$

DELTA(50)——计算各种波的理论走时震中距X的值(最多可计算50点)。

AP(50)——计算理论走时各震中距相应的初始入射角的数

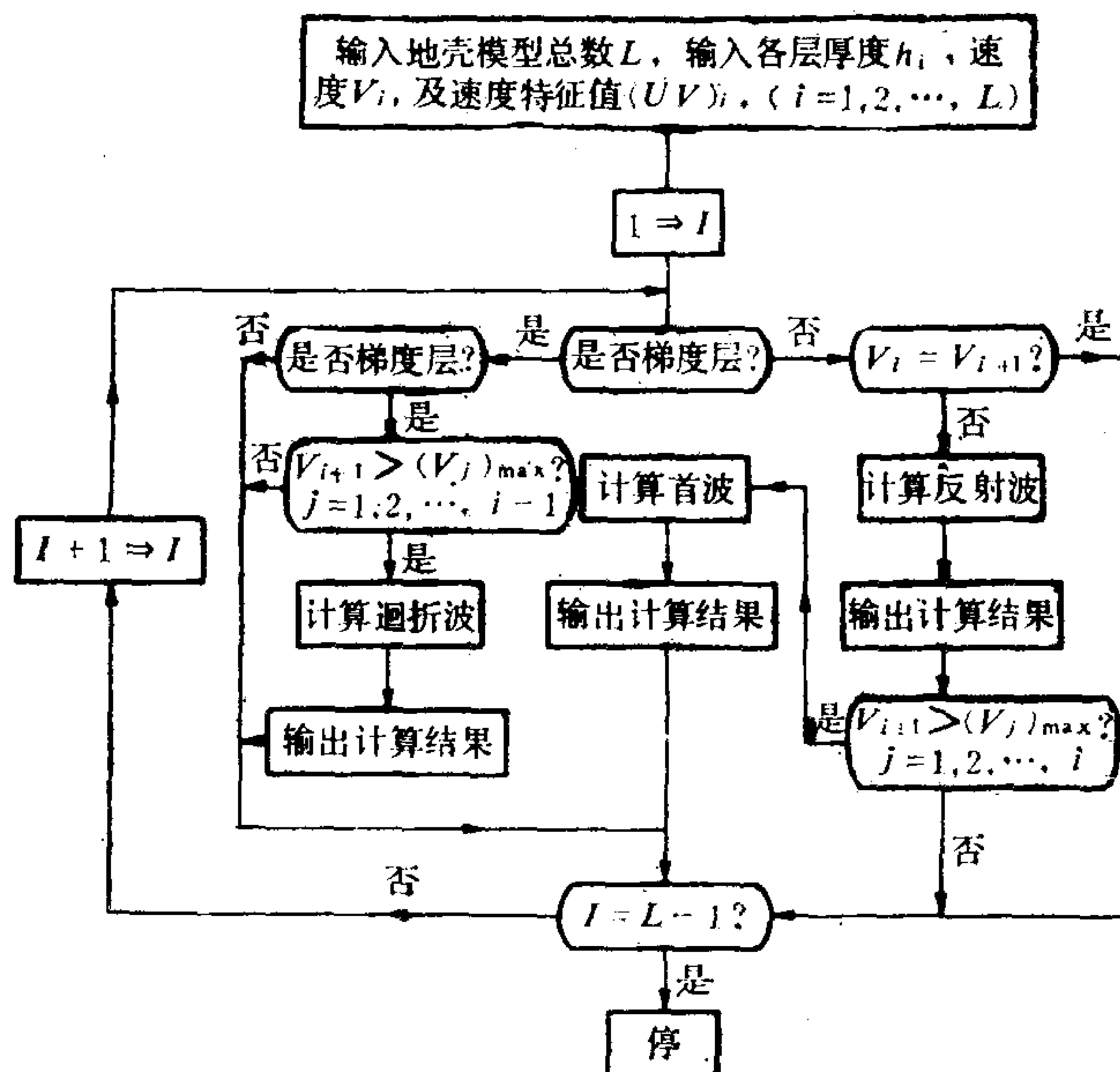



图 6-8-1

组。

T(50)——理论走时的计算值。

T1(50)——折合理论走时的计算值。$T1=T-X/6.0$（X为震中距，6.0公里/秒为地壳平均速度）。

N——计算层次变量，$N=1,2,\cdots\cdots,L-1$。

RC——计算理论走时的特征向量：

$$RC=\begin{cases}1.0 & \text{计算回折波，}\\ 0.0 & \text{计算反射波、折射波。}\end{cases}$$

PREA——调用求根子程序用的工作单元，表示初值。

DELTAO——计算回折波的临界角距离。

DELTAO——工作单元，表示已知震中距。

ROOT$(H,E,RT)$——弦位法求根子程序。其中$H$为搜索步长，$E$为求根时精度要求，$RT$为所求的根。

FAP$(A)$——函数 $F(\alpha)=X-2\sum_{i=1}^{n}x_i(\alpha)$，其中$A$表示待求的初始入射角$\alpha$。

REFT$(N,A,T)$——计算反射波走时的子程序。$N$表示层数，$A$为初始入射角，$T$为待求的走时。

RFAT$(N,A,T)$——计算回折波走时的子程序，$N$为层数，$A$为初始角，$T$为待求的走时。

**（三）FORTRAN语言源程序**

```
C       THIS IS MAIN PROGRAM
        DIMENSION THKNES(20), VP(20),AR(20),UV(20),
           DELTA(50), AP(50), T(50), T1(50)
        COMMON/TV/THKNES, VP, AR, UV/NR/N, RC,
           DELTAO/P/PREA
      1 PAUSE 00
        READ(1) L
```

```
      READ(1) (THKNES(I), VP(I), UV(I), I=1, L)
      L1=L-1
      DO 100 I=1, L1
      N=I
      IF(UV(N).EQ.0.0) GO TO 11
      AR(N)=(VP(N+1)-VP(N-1))/THKNES(N)
      IF(AR(N).LE.0.0) GO TO 100
      IF(VP(N+1).LE.VP(N-2)) GO TO 100
      RC=1.0
C     COMPUTE THE TRAVEL TIME OF REFRACTED
C     WAVE
      APA=ASIN(VP(1)/VP(N+1))
      IF(VP(N-1).LT.VP(N-2)) GO TO 21
      APB=ASIN(VP(1)/VP(N-1))
      CO TO 22
   21 APB=ASIN(VP(1)/VP(N-2))
   22 DELTAO=0.0
      DELTA(1)=-FAP(APA)
      HF=(APB-APA)/29.0
      APB=APB-HF/100.0
      DELTAO=DELTA(1)
      DELTA(30)=-FAP(APB)
      CALL RFAT(N, APA, T(1))
      CALL RFAT(N, APB, T(30))
      AP(1)=57.2957795*APA
      AP(30)=57.2957795*APB
      AP1=APA
      DO 25 J=2,29
      AP1=AP1+HF
      AP(J)=57.2957795*AP1
      DELTA(J)=-FAP(AP1)
      CALL RFAT (N, AP1, T(J))
```

```
25  CONTINUE
    DO26 J =1, 30
26  T1(J)=T(J)-DELTA(J)/6.0
    WRITE(2, 27) N
27  FORMAT(20X, I3, 20X, 14HREFRACTED WAVE)
    WRITE(2, 28)
28  FORMAT(/25X, 2HAP, 30X, 5HDELTA, 30X, 1HT,
       30X, 2HT1)
    WRITE(2, 29)(AP2(J), DELTA(J), T(J), T1(J),
       J =1, 30)
29  FORMAT(20X, F14.8, 20X, F14.6, 20X, F14.6,
       20X, F14.6)
    GO  TO 100
11  IF(VP(N).EQ.VP(N+1))  GO  TO 100
    RC =0.0
C   COMPUTE THE TRAVEL TIME OF REFLECTION
       WAVE
    PREA=0.01
    H0=1E-3
    VPMAX=1.0
    DO 12 J =1, N
12  IF(VP(J).GT.VPMAX)VPMAX=VP(J)
    AP0=ASIN(VP(1)/VPMAX)
    DELTAO=0.0
    DELTA(1)=0.0
    AP(1)=0.0
    T(1)=0.0
    DO 10 J =1, N
    IF(UV(J)=0.0)DT=THKNES(J)/VP(J)
    IF(UV(J)=1.0)DT=LN(1.0+AR(J)*THKNES(J)/
       VP(J-1)/AR(J)
10  T(1)=T(1)+DT*2.0
```

```
      WRITE(2, 16)N
16    FORMAT(20X, I3, 20X, 14HREFLECTED WAVE)
      WRITE(2, 17)
17    FORMAT(125X, 2HAP, 30X, 5HDELTA,30X, 1HT,
         30X, 2HT1)
      IF(THKNES(N)·GT·0.0)  GO  TO 31
      IF(AR(N-1)·GT·0.0)  GO  TO 19
31    DO 15 J =2, 50
      DELTAO=DELTA(J-1)+10.0
      DELTA(J)=DELTAO
      IF((PREA+H0)·GE·AP0)H0=H0/10.0
      CALL ROOT(H0, 1E-8, AP1)
      PREA=AP1
      AP(J)=57.2957795*AP1
      CALL REFT(N, AP1, T(J))
15    CONTINUE
      DO 50 J =1, 50
50    T1(J)=T(J)-DELTA(J)/6.0
      WRITE(2, 18)(AP(J), DELTA(J), T(J), T1(J),
        J =1,50)
18    FORMAT(20X,  F14.8,  20X,  F14.6,  20X, F14.,6
        20X, F14.6) GOTO 30
19    WRITE(2, 13) AP(1), DELTA(1), T(1)
13    FORMAT (20X, F14.8, 20X, F14.6, 20X, F14.6)
14    DELTAO=DELTAO+10.0
      IF(DELTAO·GE·DELA0) GO  TO  30
      IF((PREA+H0)·GE·AP0) H0=H0/10.0
      CALL  ROOT(H0, 1E-8, AP1)
      PREA=AP1
      AP2=57.2957795*AP1
      CALL REFT(N, AP1, TO)
      TO1=TO-DELTAO/6.0
```

```
      WRITE (2, 20)AP2, DELTAO, TO, TO1
   20 FORMAT(20X, F14.8, 20X,F14.6,20X, F14.6, 20X,
         F14.6) GOTO14
   30 IF(VPMAX·GE·VP(N+1)) GO TO 100
      RC=0.0
C     COMPUTE THE TRAVEL TIME OF HEAD WAVE
      AP0=ASIN(VP(1)/VP(N+1))
      DELTAO=0.0
      D1=-FAP(AP0)
      DELTA(1)=D1
      D1=D1/10.0
      M1=IFIX(D1)
      D1=FLOAT(M1)
      D1=D1*10.0
      CALL REFT(N, AP0, T(1))
      DO 35 J=2, 50
      D1=D1+10.0
      DELTA(J)=D1
   35 T(J)=T(1)+(D1-DELTA(1))/VP(N+1)
      DO 39 J=1, 50
   39 T1(J)=T(J)-DELTA(J)/6.0
      WRITE(2, 36)N
   36 FORMAT(20X, I3, 20X, 9HHEAD WAVE)
      WRITE (2, 37)
   37 FORMAT(/25X, 5HDELTA, 30X,1HT,30X, 2HT1)
      WRITE(2, 38)(DELTA(J), T(J), T1(J), J=1, 50)
   38 FORMAT(20X, F14.6, 20X, F14.6, 20X, F14.6)
  100 CONTINUE
      GO TO 1
      END
C     SUBPROGRAM TO FIND ROOTS BY ITERATIVE
C     METHOD
```

```
      SUBROUTINE ROOT(H, E, RT)
      COMMON/P/PREA
      W=0.0
      X=PREA
      F0=FAP(X)
4     X=X+H
      F1=FAP(X)
      IF(F1/F0.GT.0.0) GO TO 101
      FA=F0
      FB=F1
      A=X-H
      B=X
1     C=B-(A-B)*FB/(FA-FB)
      IF(W)5, 7, 5
7     W=1.0
      GO TO 6
5     IF(ABS(C1-C).LT.E) GO TO 3
6     C1=C
      FC=FAP(C)
      IF(FC*FA.GT.0.0)GOTO2
      FB=FC
      B=C
      GOTO 1
2     FA=FC
      A=C
      GOTO 1
3     RT=C
      GO TO 102
101   F0=F1
      IF(X-10.0)4, 102, 102
102   RETURN
      END
```

```
      SUBPROGRAM TO COMPUTE INITIAL VALUE
      OF INCIDENT ANGLE BY EPICENTRAL DISTAN
         CE FUNCTION FAP(A)
      DIMENSION THKNES(20),VP(20),AR(20), UV(20)
      COMMON/TV/THKNES, VP, AR, UV/NR/N, RC,
         DELTAO
      ALPHA=A
      P=SIN(ALPHA)/VP(1)
      S=0.0
      DO 100 I=1, N
      IF(UV(I).NE.0.0) GO TO 20
      B=P*VP(I)
      DS=THKNES(I)*B/SQRT(1.0-B*B)
      GOTO 10
   20 AR(I)=(VP(I+1)-VP(I-1))/THKNES(I)
      IF(B.GE.1.0) GO TO 100
      APHA=ASIN(B)
      IF(I.NE.N)GO TO 15
      BR=RC*AR(I)
      IF(BR.GT.0.0) GO TO 30
      BB=P*VP(I+1)
     IF(BB.GE.1.0) GO TO 100
   15 APHB=ASIN(BB)
      DS=(COS(APHA)-COS(APHB))/(P*AR(I))
      GOTO 10
   30 DS=COS(APHA)/(P*AR(I))
   10 S=S+2.0*DS
  100 CONTINUE
      FAP=DELTAO-S
      RETURN
      END
C     SUB PROGRAM TO COMPUTE TRAVEL TIME
C     OF THE REFLECTED WAVE
```

```
      SUB ROUTINE REFT(N, A, T)
      DIMENSION THKNES(20), VP(20), AR(20), UV(20)
      COMMON/TV/THKNES, VP, AR, UV
      P=SIN(A)/VP(1)
      T=0.0
      DO 20J=1, N
      IF(UV(J)·NE·0.0)  GO  TO 10
      B=P*VP(J)
      DT=THKNES(J)/(VP(J)*SQRT(1.0-B*B))
      GO  TO 20
10    APHA=ASIN(P*VP(J-1))
      APHB=ASIN(P*VP(J+1))
      HA=TAN(APHA/2.0)
      HB=TAM(APHB/2.0)
      DT=ALOG(HB/HA)/AR(J)
20    T=T+2.0*DT
      RETURN
      END
C       SUBPROGRAM TO COMPUTE TRAVEL TIME
C       OF THE
C       REFRACTED WAVE
      SUBROUTINE RFAT(N, A, T)
      DIMENSION THKNES(20), VP(20), AR(20), UV(20)
      COMMON/TV/THKNES, VP, AR, UV
      ALPHA=A
      P=SIN(ALPHA)/VP(1)
      N1=N-1
      CALL REFT(N1, ALPHA, T1)
      B=P*VP(N-1)
      IF(B·GE·1.0)  GO  TO 10
      ALPHB=ASIN(B)
      HF=1.0/TAN(ALPHB/2.0)
```

```
      DT=ALOG(HF)/AR(N)
      GO TO 20
10    DT=0.0
20    T=T1+20*DT
      RETURN
      END
```

## （二）用反射波走时反演地壳结构程序

本程序利用观测的反射波走时反演地壳结构。假定层状介质的分界面是水平的，对各个反射界面自上而下逐层进行计算，可以获得各层的厚度 $h_i$ 及波速 $V_i$。计算过程及框图已在前面讨论过，不再赘述。

**（一）主要标识符及子程序说明**

THKO(20)——层状介质初始模型的各层厚度 $h_i$（最多计算20层）。

VPO(20)——初始模型各层速度$V_i$。

DLT(100)——实测走时曲线各点震中距（最多 100 个观测点）。

OBS(100)——实测走时曲线各点观测走时。

F(100)——由初始模型计算的理论走时。

*VH*(40)——由各层参数$V_i$及$h_i$组成的参数数组，奇数部分为$V_i$，偶数部分为$h_i$。

L1(40)——表示在反演过程中参数是否修定的数组。当

$$L1[I]=\begin{cases}1 & \text{第 I 个参数要修定，}\\0 & \text{第 I 个参数不修定。}\end{cases}$$

L2(20)——表示某层为常速层还是速度梯度层。

$$L2(I)=\begin{cases}1 & \text{第 I 层为速度梯度层，}\\0 & \text{第 I 层为常速层。}\end{cases}$$

N——地壳模型的分层数。

M——反射波走时曲线的测点数。

AR(20)——速度梯度层的梯度值$a_i = \frac{V_{i+1} - V_{i-1}}{h_i}$。

AP1(100)——相应于各测点震中距相应的射线初始入射角。

PREA——迭代法求根的初值。

AC(100, 100)——Jacobi矩阵中的微分系数$\frac{\partial f_i}{\partial x_i}$。

TD(100)——计算Jacobi矩阵的工作单元。

DEF(100)——观测到时与理论到时残差$\Delta t_i$。

AA(40, 40)——正规方程组系数矩阵$A^T A$中的各元素。

G(40)——正规方程组右端项$A^T \Delta t$各元素。

DX(40)——参数修正量$\Delta X$各元素。

OLDX(40), $P$(40)——工作单元。

REFL(X)——计算反射波理论走时的子程序，虚元$X$为层参数(速度及厚度)。

FAP(A)——根据已知震中距求反射波初始入射角$\alpha$的函数$F(\alpha)$的子程序。虚元$A$表示入射角。

ROOT(H,E,RT)——用线性插值法求非线性方程的根的子程序。虚元意义同附录一。

DERIVE(X)——形成Jacobi矩阵（微分系数矩阵）的子程序。虚元$X$为层参数。

CALPHI(PHI)——计算残差平方和的子程序。虚元PHI为走时残差平方和$\sum_{i=1}^{m} \Delta t_i^2$。

LDLT(N,A,P,B,X,ISW)——用改进平方根法(Cholesky法)解线性方程组的子程序。虚元的意义为：

N——方程的阶数；

A——方程组的系数矩阵；

P——工作单元；

B——方程组的右端系数项；

X——方程组的解；

ISW——若方阵A非对称正定或奇异，解方程失败，ISW = 0，并返回。否则，ISW = 1，按正常计算步骤算出结果。

MAQT(X,PLMDO,KENN,S,ITER,N1)——用阻尼最小二乘法求目标函数极小的子程序。各虚元的意义为：

N1——待修改的参数个数；

PLMDO——阻尼系数$p$的初值；

KENN——整型量，用以标志搜索目标函数极小时迭代过程结束的原因。其中

KENN = 0　迭代过程在满足精度要求时结束；

KENN = 1　迭代过程已达到最大次数（本程序取30次）而结束；

KENN = −1　标志对入搜索$S(x+\lambda\Delta x)$极小的过程已失效；

KENN = −2　标志求X时解正定对称方程组时失效。

S——目标函数（残差平方和）的值。

ITER——迭代次数、整型量。

（二）FORTRAN-77语言源程序：

```
C       THIS PROGRAM IS DESIGNED FOR TRAVEL
C       TIME INVERSE OF
C       REFLECTED WAVE IN MULTI-LAYERS
C
        DIMENSION THKO(20), VPO(20), VH(40)
        COMMON/NN/N, M
        COMMON/O/OBS(100)
        COMMON/FF/F(100)
        COMMON/D/DLT(100)
```

```
      COMMON/LL/L1(40), L2(20)
      PRINT *,'DATA N, M, DLT, OBS, THKO, VPO,
        L1 AND L2 ARE WANTED NOW'.
      PAUSE 1
      READ *, N, M
      PAUSE 2
      READ(*, *) (DLT(I), I=1, M)
      PAUSE 3
      READ(*, *) (OBS(I), I=1, M)
      PAUSE 4
      READ(*, *)(THKO(I), I=1, N)
      PAUSE 5
      READ(*, *)(VPO(I), I=1, N)
      PAUSE 6
      READ(*, *)(L1(I), I=1, 2*N)
      PAUSE 7
      READ(*, *)(L2(I), I=1, N)
      DO 10 I=1, N
      VH(2*I-1)=VPO(I)
10    VH(2*I)=THKO(I)
      N1=0
      DO 20 I=1, 2*N
      IF(L1(I).NE.0)N1=N1+1
20    CONTINUE
      CALL MAQT(VH, 1.0, KN, S, ITR, N1)
      WRITE(*, 30)KN, S, ITR
30    FORMAT(//,10X'THE COMPUTING PARAMETER
        S',
        //,5X,'KN=',I2'5X, 'S=',F10.8,5X,'ITR=',13)
      WRITE(*, 40)(DLT(I), F(I), OBS(I), I=1, M)
40    FORMAT(/, 5X,'D=',F8.3, 8X,'TF=', F8.4,8X,
        'TO=', F8.4)
```

```
      WRITE(*, 50)(VH(2*I-1), VH(2*I), I=1, N)
 50   FORMAT(//, 10X, 'VIN=', F6.3, 10X, 'HIN'=,
         F8.3)
      WRITE(*, 60)(VPO(I), THKO(I), I=1, N)
 60   FORMAT(//, 10X, 'VPR=', F6.3, 10X, 'HPR=',
         F8.3)
      STOP
      END
C
      SUBROUTINE REFL(X)
      DIMENSION X(100), AP1(100)
      COMMON/NN/N, M
      COMMON/T/THKS(20)
      COMMON/D/DLT(100)
      COMMON/V/VP(20)
      COMMON/FF/F(100)
      COMMON/PP/PREA
      COMMON/DT/DLTO
      COMMON/GR/AR(20)
      COMMON/LL/L1(40), L2(20)
      DO 10 I=1, N
      VP(I)=X(2*I-1)
 10   THKS(I)=X(2*I)
      H0=0.005
      VPM=1.0
      DO 20 J=1, N
      IF(VP(J).GT.VPM)VPM=VP(J)
 20   CONTINUE
      AP0=ASIN(VP(1)/VPM)
      PREA=0.01
      DO 100 I=1, M
      DLTO=DLT(I)
```

```
      IF((PREA+H0).GE.AP0)H0=H0/10.0
      CALL ROOT (H0, 1E-4, AP1(I))
      PREA=AP1(I)
      P=SIN(AP1(I))/VP(1)
      F(I)=0.0
      DO 30 J=1, N
      IF(L2(J).NE.0)  GO  TO 40
      H=P*VP(J)
      IF(H.GT.1.0)  GO  TO 100
      C=SQRT(1.0-H*H)
      DTM=THKS(J)/(VP(J)*C)
      GOTO 30
40    PV=P*VP(J)
      PV1=P*VP(J+1)
      IF(PV.GT.1.0.OR.PV1.GT.1.0)  GO  TO 100
      APA=ASIN(PV)
      APB=ASIN(PV1)
      HA=TAN(APA/2.0)
      HB=TAN(APB/2.0)
      DTM=ALOG(HB/HA)/AR(J)
30    F(I)=F(I)+2.0*DTM
100   CONTINUE
      RETURN
      END
C
      SUBROUTINE ROOT(H, E, RT)
      COMMON/PP/PREA
      W=0.0
      X=PREA
      F0=FAP(X)
40    X=X+H
      F1=FAP(X)
```

```
      IF(F1/F0.GT.0.0)  GO  TO 101
      FA=F0
      FB=F1
      A=X-H
      B=X
   10 C=B-(A-B)*FB/(FA-FB)
      IF(W)50, 70, 50
   70 W=1.0
      GO  TO 60
   50 IF(ABS(C1-C), LT, E)  GO  TO 30
   60 C1=C
      FC=FAP(C)
      IF(FC*FA.GT.0.0) GO  TO 20
      FB=FC
      B=C
      GO  TO 10
   20 FA=FC
      A=C
      GO  TO 10
   30 RT=C
      GO  TO 102
  101 F0=F1
      IF(X-10.0)40, 102, 102
  102 RETURN
      END
C
      FUNCTION FAP(A)
      COMMON/NN/N, M
      COMMON/T/THKS(20)
      COMMON/V/VP(20)
      COMMON/DT/DLTO
      COMMON/GR/AR(20)
```

```
      COMMON/LL/L1(40), L2(20)
      ALP=A
      P=SIN(ALP)/VP(1)
      S=0.
      DO 100 I=1, N
      IF (L2(I).NE.0)  GO  TO 20
      D=P*VP(I)
      IF(D.GT.1.0)  GO  TO 100
      DS=THKS(I)*D/SQRT(1.0-D*D)
      GOTO 10
   20 AR(I)=(VP(I+1)-VP(I))/THKS(I)
      PV=P*VP(I)
      PV1=P*VP(I+1)
      IF(PV.GT.1.0).OR.PV1.GT.1.0)  GO  TO 100
      APA=ASIN(PV)
      APB=ASIN(PV1)
      DS=(COS(APA)-COS(APB))/(P*AR(I))
   10 S=S+2.0*DS
  100 CONTINUE
      FAP=S-DLTO
      RETURN
      END
C
      SUBROUTINE LDLT(N, A, P, B, X, ISW)
      DIMENSION A(40,40), P(40), B(40), X(40)
      DO 5 I=1, N
      DO 5 J=1, I
      W=A(J, I)
      IF(I-J)10, 20, 10
   20 IF(J.EQ.1)  GO  TO 30
      DO 25 L=2, J
      K=J-L+1
```

```
      Y=A(I, K)
      Z=Y*P(K)
      A(I, K)=Z
   25 W=W-Y*Z
   30 IF(W.NE.0.0) GO TO 35
      ISW=0
      RETURN
   35 P(I)=1.0/W
      GOTO 5
   10 IF(J.EQ.1) GO TO 45
      DO 40 L=2, J
      K=J-L+1
   40 W=W-A(I, K)*A(J, K)
   45 A(I, J)=W
    5 CONTINUE
      DO 60 I=1, N
      Y=B(I)
      IF(I.EQ.1) GO TO 60
      DO 65 L=2, I
      K=I-L+1
   65 Y=Y-A(I, K)*X(K)
   60 X(I)=Y
      DO 100 L=1, N
      I=N-L+1
      Y=X(I)*P(I)
      IF(I.EQ.N) GO TO 100
      IA1=I+1
      DO 70 K=IA1, N
   70 Y=Y-A(K, I)*X(K)
  100 X(I)=Y
      ISW=1
      RETURN
```

```
      END
C
      SUBROUTINE DERIVE(X)
      DIMENSION X(100), TD(100)
      COMMON/NN/N, M
      COMMON/FF/F(100)
      COMMON/ACC/AC(100, 100)
      COMMON/LL/L1(40), L2(20)
      CALL REFL(X)
      DO 10 I=1, M
   10 TD(I)=F(I)
      K1=0
      DO 101 I=1, 2*N
      IF(L1(I).EQ.0) GO TO 100
      K1=K1+1
      X(I)=X(I)+1E-5
      CALL REFL(X)
      X(I)=X(I)-1E-5
      DO 105 J=1, M
  105 AC(J, K1)=(F(J)-TD(J))/1E-5
  100 CONTINUE
  101 CONTINUE
      RETURN
      END
C
      SUBROUTINE CALPHI(PHI)
      COMMON/NN/N, M
      COMMON/O/OBS(100)
      COMMON/FF/F(100)
      COMMON/DE/DEF(100)
      PHI=0
      DO 100 I=1, M
```

```
      DEF(I)=OBS(I)-F(I)
      PHI=PHI+DEF(I)**2
100   CONTINUE
      RETURN
      END
C
C
      SUBROUTINE MAQT(X, PLDO, KN, S, ITR, N1)
      DIMENSION X(40), AA(40, 40), G(40), DX(40),
         OLDX(40), P(40)
      COMMON/NN/N, M
      COMMON/ACC/AC(100, 100)
      COMMON/DE/DEF(100)
      COMMON/FF/F(100)
      COMMON/LL/L1(40), L2(20)
      ITR=0
      PLD=PLDO
      ISB=1
      KN=0
      GO TO 1000
1111  PHIO=PHI
  10  ITR=ITR+1
      WRITE(*, 11)ITR, PLD
  11  FORMAT(/, 2X, 'ITR=', I3,10X,'PLD=',F6.2)
      IF(ITR.LE.30) GO TO 30
      KN=1
      GO TO 10000
  30  CALL DERIVE(X)
      DO 60 I=1, N1
      HF=0.0
      DO 70 K=1, M
  70  HF=HF+AC(K, I)*DEF(K)
```

```
      G(I)=HF
      DX(I)=HF
      DO 80 K=I, N1
      HF=0.0
      DO 90 J=1, M
   90 HF=HF+AC(J, I)*AC(J, K)
      AA(I, K)=HF
   80 AA(K, I)=HF
   60 AA(I, I)=AA(I, I)+PLD
      CALL LDLT(N1, AA, P, G, DX, ISW)
      IF(ISW.EQ.1) GO TO 100
      KN=-2
      GO TO 10000
  100 K1=0
      DO 120 I=1, 2*N
      OLDX(I)=X(I)
      IF(L1(I).EQ.0.0)GO TO 120
      K1=K1+1
      X(I)=X(I)+DX(K1)
  120 CONTINUE
      ISB=2
      GO TO 1000
 2222 WRITE(*, 20)PHI, PHIO
   20 FORMAT(/, 10X, 'PHI=', F14.8,5X, 'PHIO=',
         F14.8)
      IF(PHI.GE.PHIO) GO TO 130
      C=1.0
      GO TO 140
  130 GDX=0.0
      DO 160 I=1, N1
      GDX=GDX+G(I)*DX(I)
  160 CONTINUE
```

```
      C=GDX/(PHI-PHIO+2.0*GDX)
      L=0
180   L=L+1
      K1=0
      DO 190 I=1, 2*N
      IF(L1(I).EQ.0)  GO  TO 190
      K1=K1+1
      X(I)=OLDX(I)+C*DX(K1)
190   CONTINUE
      GO  TO 1000
3333  IF(PHI.LT.PHIO)  GO  TO 140
      IF(L.LT.7)  GO  TO 210
      KN=-1
      DO 220 I=1, N
220   X(I)=OLDX(I)
      PLD=1.5*PLD
      GO TO 10
210   C=.5*C
      GO TO 180
140   HF=0.0
      HZ=0.0
      DO 230 I=1, N1
      HF=HF+ABS(C*DX(I))
230   HZ=HZ+ABS(X(I))
      IF(HF.LT.1E-3*HZ)  GO  TO 10000
      IF(PHI.LT.1E-3)  GO  TO 10000
      PHIO=PHI
      PLD=0.7*PLD
      GO TO 10
1000  CALL REFL(X)
      WRITE(*, 5000)(F(I), I=1, M)
5000  FORMAT(/1X, 10F7.2)
```

```
      WRITE(*, 6000)(X(I), I=1, 2*N)
6000  FORMAT(/, 5X, 'XI=', 2F18.7)
      CALL CALPHI(PHI)
      GO TO(1111, 2222, 3333), ISB
10000 S=PHI
      RETURN
      END
```

# 第七章　用地震体波反演地球内部速度分布

地震学经典问题之一是由地震体波的走时反演地球内部的速度分布。在本世纪初，由于地震台网的建立和观测资料的逐步积累，先后由 Oldham (1906)，Zöeppritz (1907)，Herglotz (1907)，Wiechert (1910)，Mohorovičič (1909)及Gutenberg (1915)等人作了重要的奠基工作，初步建立了地球内部层状结构模型。到了本世纪四十年代，由于 Jeffres 及 Bullen 等人的工作，得以建立起至今还适用的地球速度模型。五十年代以来，由于地震台网的加密和对地下核爆炸观测工作的开展，世界许多地区通过体波走时研究，发现地慢内速度分布有较大的横向变化。几个主要特征层(低速层，速度梯度层)的深度和速度值各地区也有不少差异。这种情况为研究地球内部结构，特别是为研究上地幔的结构特征提供了重要资料。

这一章，我们讨论用 $P$波走时反演上地幔速度分布的计算问题。

## §7-1　球对称分层介质中的地震射线

我们考虑一个球对称地球模型，其内部速度分布是地球半径的函数。

为了推导地震射线的路径，先来考虑一个均匀同心球层模型，如图7-1-1所示。设各层的波速分别为 $V_1$，$V_2$，…。由$P_0$点发出的地震射线在各层之间的界面上将发生折射。对于第一个分

界面，由Snell定律有：

$$\frac{\sin i_1}{V_1}=\frac{\sin\alpha}{V_2} .\qquad (7\text{-}1.1)$$

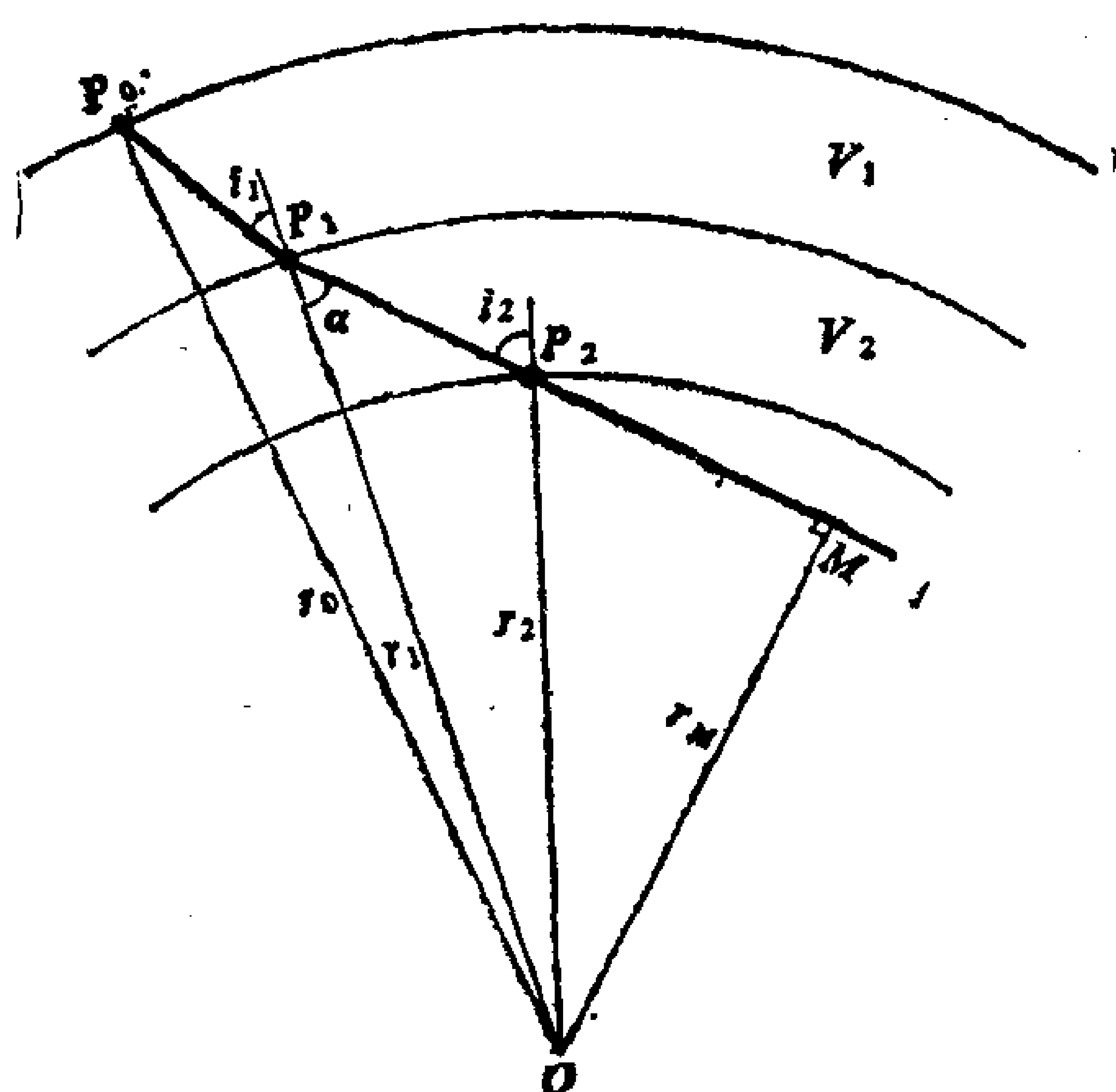



图 7-1-1

在三角形$OP_1P_2$中应用正弦定理，

$$\frac{\sin\alpha}{r_2}=\frac{\sin i_2}{r_1} ,\qquad (7\text{-}1.2)$$

由(7-1.1)及(7—1.2)式消去$\sin\alpha$，则得

$$\frac{r_1\sin i_1}{V_1}=\frac{r_2\sin i_2}{V_2} .\qquad (7\text{-}1.3)$$

当增加到$n$层时．可以写出

$$\frac{r_1\sin i_1}{V_1}=\frac{r_2\sin i_2}{V_2}=\cdots=\frac{r_n\sin i_n}{V_n} .\qquad (7\text{-}1.4)$$

这是球对称层状介质中Snell定律的形式。若球层划分无限增加，则速度随$r$连续变化，$V=V(r)$。这时射线是一条光滑的连续曲线，仍有如下关系：

$$\frac{r\sin i}{V(r)}=\frac{r_0\sin i_0}{V_0}=p. \tag{7-1.5}$$

式中$r_0$为地球半径，$i_0$及$V_0$分别为地表处的入射角及波速，如图7-1-2所示。由于$r_0$、$i_0$、$V_0$对于某一射线而言是常数，因此$p$也是常数，称为射线参数，其单位为秒/弧度〔若取0.0174533 $p$，其单位为秒/度，若取(0.0174533/111.195) $p$，其单位则为秒/公里〕。这就是球对称连续介质中的折射定律。

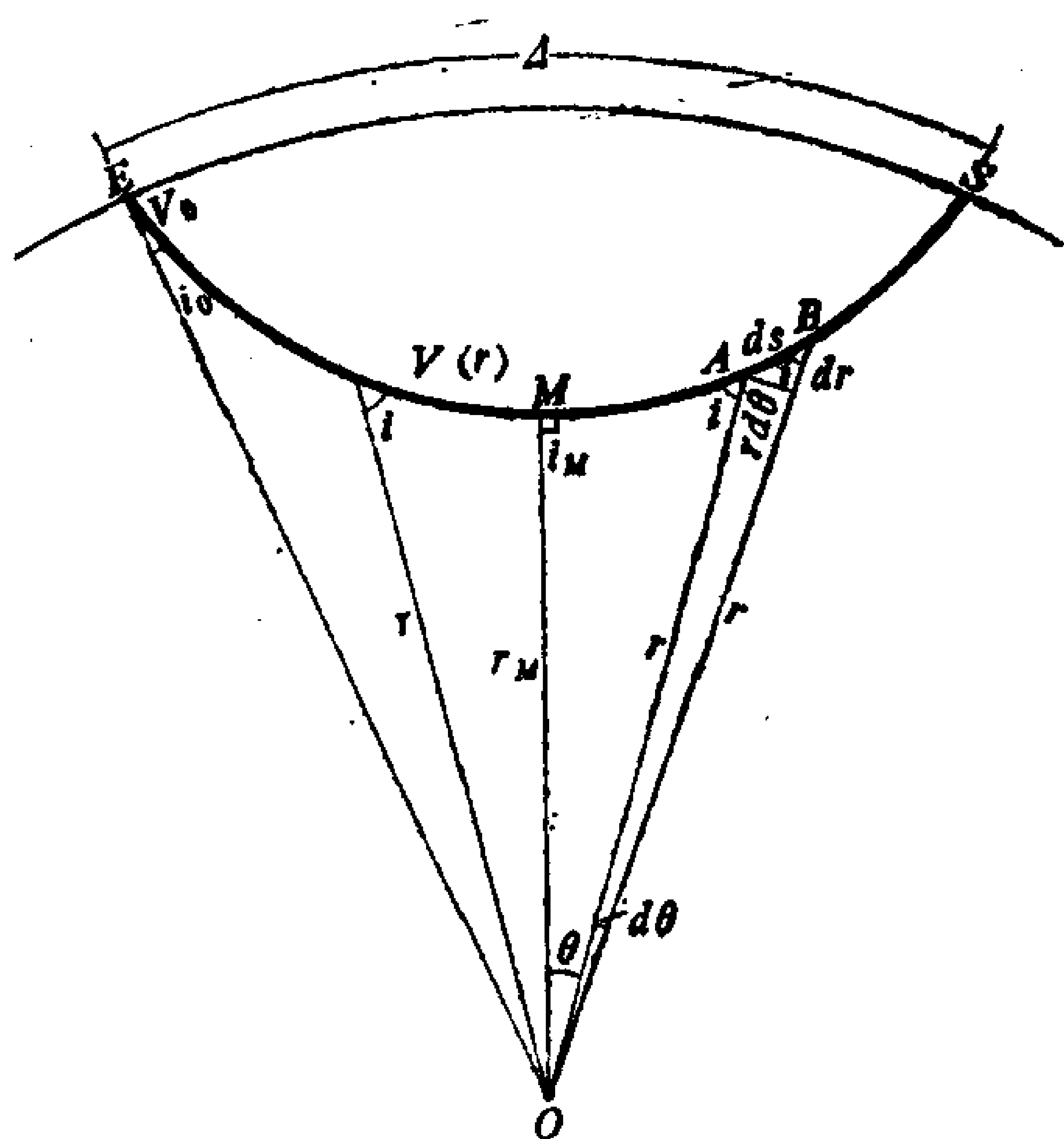



图 7-1-2

由图 7-1-2 见，当射线处于最低点$M$时（对应的半径$r_M$为最小），其入射角（折射角）$i_M$必等于90°，故有

$$\frac{r_M}{V(r_M)}=p\ . \tag{7-1.6}$$

因此，对于一定的速度分布的球对称介质，射线穿透的深度取决于射线参数$p$。换言之，具有某一射线参数的射线，其穿透深度也随之固定下来。

下面来推导射线方程，即某一射线通过球对称介质的路径及走时，对于图7-1-2中的射线$EMS$，考虑射线上无限相邻两点$A$、$B$，令$AB$段射线弧长为$ds$，不难看出有

$$ds^2=dr^2+r^2d\theta^2, \tag{7-1.7}$$

及
$$\sin i=\frac{rd\theta}{ds}\ ,\qquad \cos i=\frac{dr}{ds}\ . \tag{7-1.8}$$

由(7-1.8)式得

$$d\theta=\pm\frac{\sin i}{\cos i}\ \frac{dr}{r}. \qquad (7\text{-}1.9)$$

再由(7-1.5)式得，$\sin i=\frac{V(r)}{r}p$，代入上式得

$$d\theta=\pm\frac{p}{r\sqrt{\frac{r^2}{V^2(r)}-p^2}}dr. \qquad (7\text{-}1.10)$$

这就是连续介质中的射线微分方程。正负号对应于射线最低点$M$两侧的不同段。对$EM$段取负号，$MS$段取正号。对(7-1.10)式积分得出$\theta$与$r$的关系

$$\theta(r)=\pm\int_{r_M}^{r}\frac{pdr}{r\sqrt{\frac{r^2}{V^2(r)}-p^2}}. \qquad (7\text{-}1.11)$$

这称为射 线方程。其中 $r_M$ 满足 $p=\frac{r_M}{V(r_M)}$，当$r=r_M$时，$\theta(r_M)=0$，即$\theta$角是从射线最低点处起算。

现在来推导走时方程。地震波由$A$传 到 $B$ 所需的时间为（图 7-1-2）

$$dt=\frac{ds}{V}, \qquad (7\text{-}1.12)$$

将(7-1.7)式$ds$代入上式，并考虑到(7-1.10)式，可以得

$$dt=\pm\frac{rdr}{V^2(r)\sqrt{\frac{r^2}{V^2(r)}-p^2}}. \qquad (7\text{-}1.13)$$

如果令图 7-1-2 中整个射线 $ES$ 对 应 的 弧 度 角为 $\Delta$，则对(7-1.10)及(7-1.13)分别积分得

$$\Delta=2p\int_{r_M}^{R}\frac{dr}{r\sqrt{\frac{r^2}{V^2(r)}-p^2}},$$

$$T = 2\int_{r_M}^{R} \frac{r dr}{V^2(r)\sqrt{\frac{r^2}{V^2(r)} - p^2}}. \qquad (7\text{-}1.14)$$

上式中 $R$为地球的平均半径。这就是地震射线在球对称连续介质中的走时方程。它们均以射线参数 $p$ 为参数。可见，对于一定入射角为$i$（其射线参数$p$也随之确定）的射线，其传播距离$\Delta$及走时$T$均是确定的。

为了能从地面实测的走时曲线来求出射线参数 $p$值，现在来研究走时曲线的斜率 $dt/d\Delta$与射线参数$p$的关系。如图7-1-3所示，设有由$E$出发，在同一平面内的两条相邻射线 $EA$及$EB$、$AB$之间的距离为$d\Delta$（以弧度为单位），$AA'$ 为波前。经过$dT$时间后，波前传至$B$处。设地表处的波速为$V_0$，则有

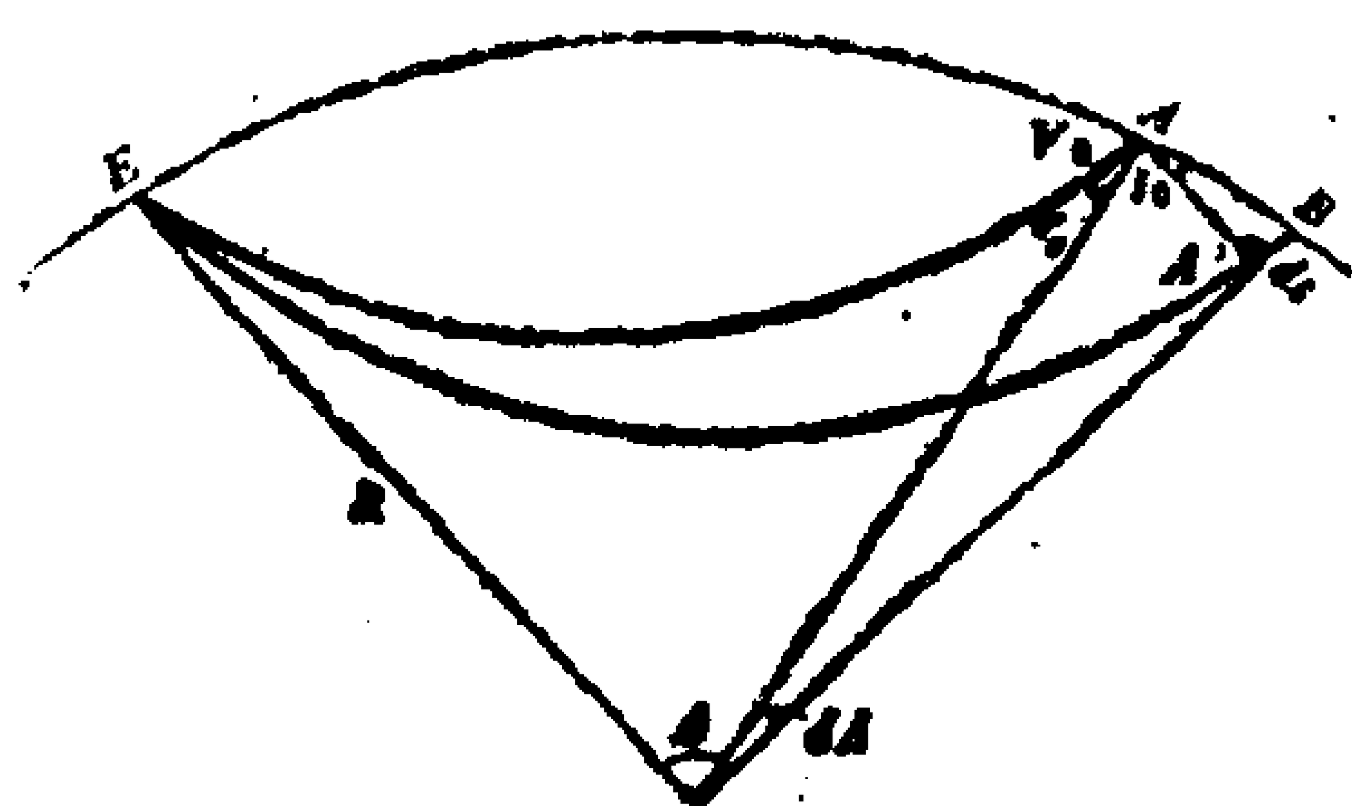



图 7-1-3

$$A'B = ds = V_0 dT. \qquad (7\text{-}1.15)$$

由图7-1-3几何关系可知

$$\frac{ds}{R d\Delta} = \sin i_0 , \qquad (7\text{-}1.16)$$

将(7-1.5)式及(7-1.15)式代入则得

$$\frac{dT}{d\Delta} = \frac{R \sin i_0}{V_0} = p. \qquad (7\text{-}1.17)$$

由此可见，走时曲线上某一点的斜率 $dT/d\Delta$（又称慢度，单位为秒/度或秒/弧度），就等于到达该点射线的射线参数 $p$。这是一个很重要的结果。这意味着从实测的走时曲线上可以直接求出参数$p$，这对今后计算很有用。

再者，由图7-1-3可知

$$\frac{Rd\Delta}{dT}=\overline{V}_0 \qquad \text{为地震波的视速度;}$$

$$\frac{ds}{dT}=V_0 \qquad \text{为地震波的真速度.}$$

考虑到(7-1.16)式得

$$\sin i_0=\frac{V_0}{\overline{V}_0} \quad . \tag{7-1.18}$$

上式表明，地震波的视速度取决于射线的入射角 $i_0$。表示地震波视速度与真速度关系的(7-1.18)式，又称为Bendoff定律。

## §7-2 地球内部的速度分布对地震射线及走时的影响

地球内部体波传播的路径基本取决于介质的速度分布状况。下面，我们要讨论几种典型的速度变化情况对地震射线及走时的影响。识别射线和走时的这些特征，对反演地球内部速度分布是有用的。

一般说来，要使球对称地球中的射线返回地面，其内部速度分布必须满足条件

$$\frac{dV}{dr}<\frac{V}{r} \quad . \tag{7-2.1}$$

下面分别几种情况进行讨论：

### （一）速度连续变化的情况

(1) $\frac{dV}{dr}<0$ 这时速度随深度增加(半径$r$减小)而增加，每一条射线都有一个最低点 而且都是向上弯曲的(7-2-1(a))。因此全部射线都出射到地表。其走时曲线如图7-2-1(b)示。

(2) $\frac{dV}{dr}=0$ 这时速度为常量$V$,射线为直线(7-2-2(a))

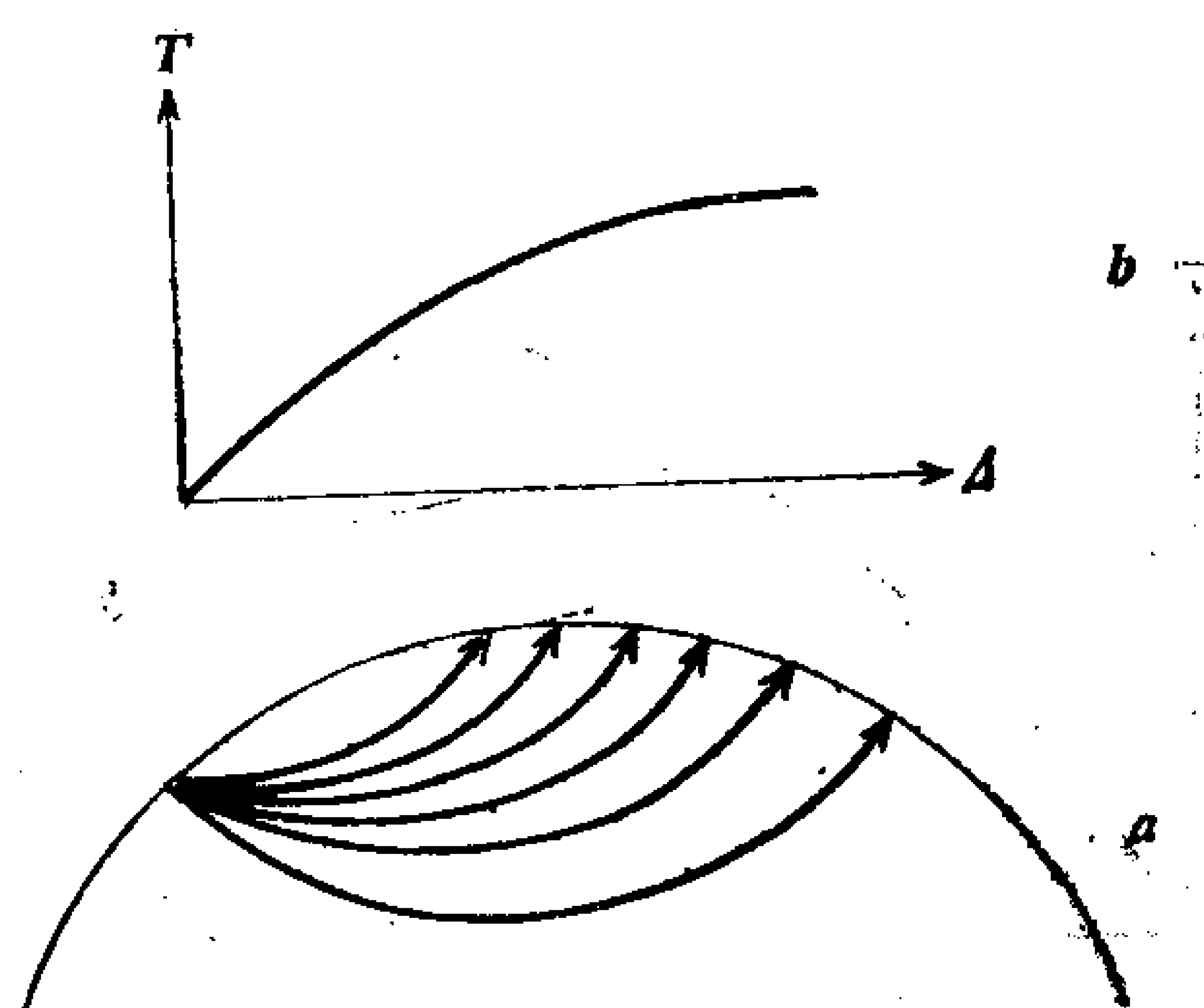



图 7-2-1

走时$T$与距离$\Delta$满足关系

$$T = \frac{2R\sin\frac{\Delta}{2}}{V}\quad (7\text{-}2.2)$$

因而走时曲线呈现为正弦曲线，如图7-2-2(b)示。

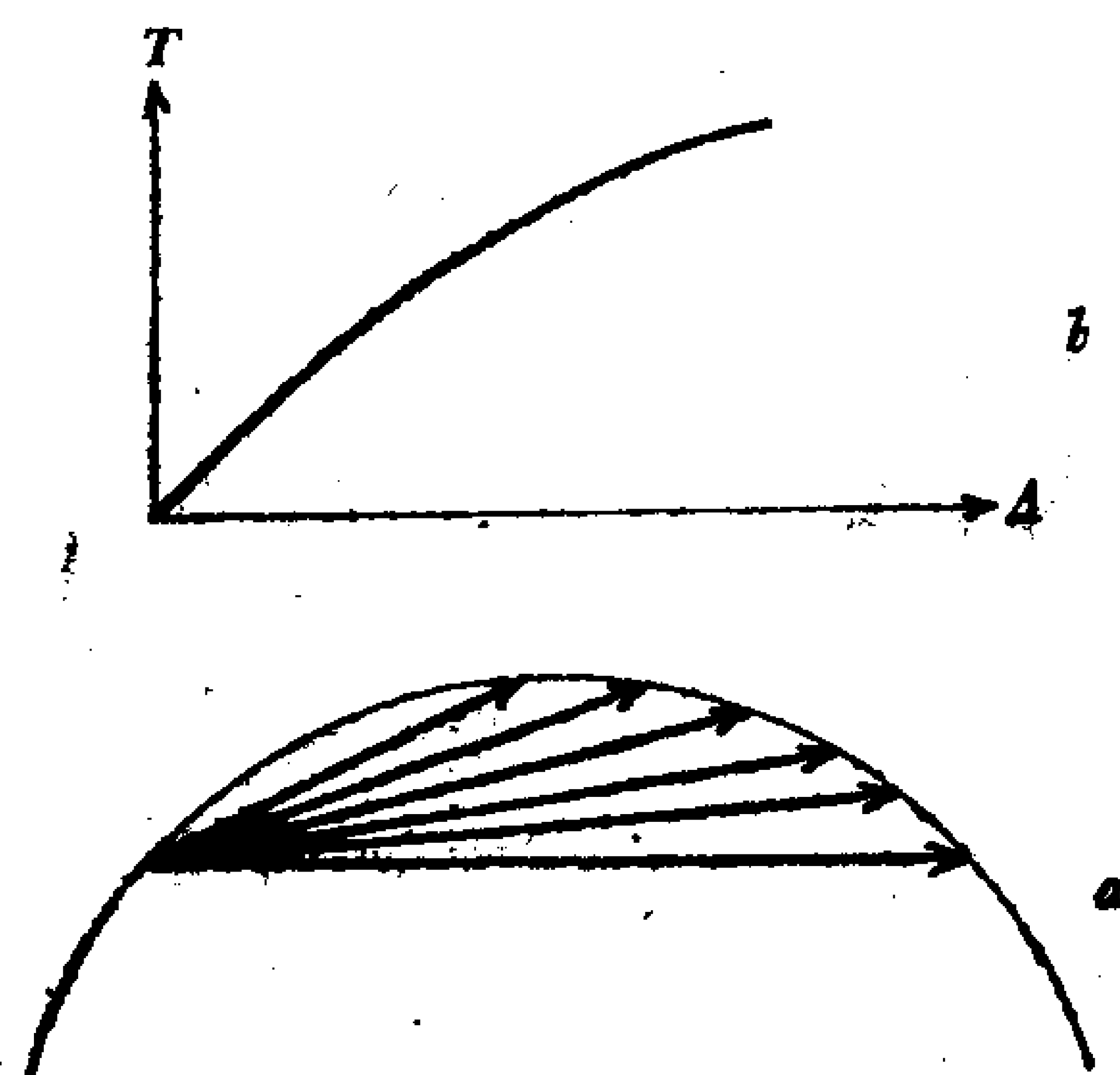



图 7-2-2

(3) $\frac{dV}{dr}>0$ 这种情况相当于速度值随深度增大而减小。若其减小率符合条件(7-2.1)，则地震射线为向下弯曲且射

线曲率小于地表曲率的曲线，能出射到地表(但视速度很低)如图7-2-3(a)示。

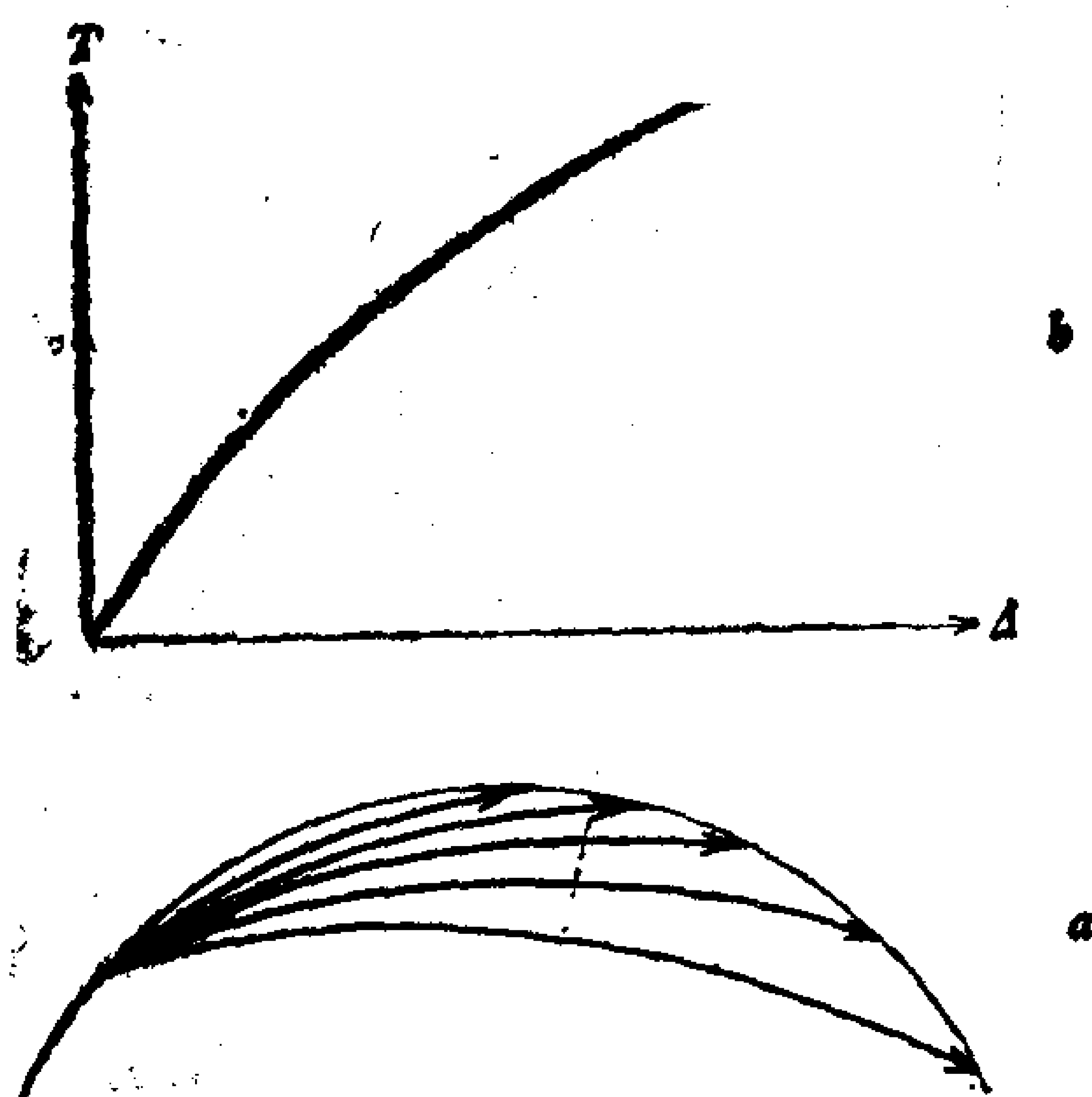



图 7-2-3

当速度减小率不满足(7-2.1)式时，除$i_0=90°$的射线沿地面传播外，其余全部射线呈螺旋线卷入地心，如图7-2-4所示。

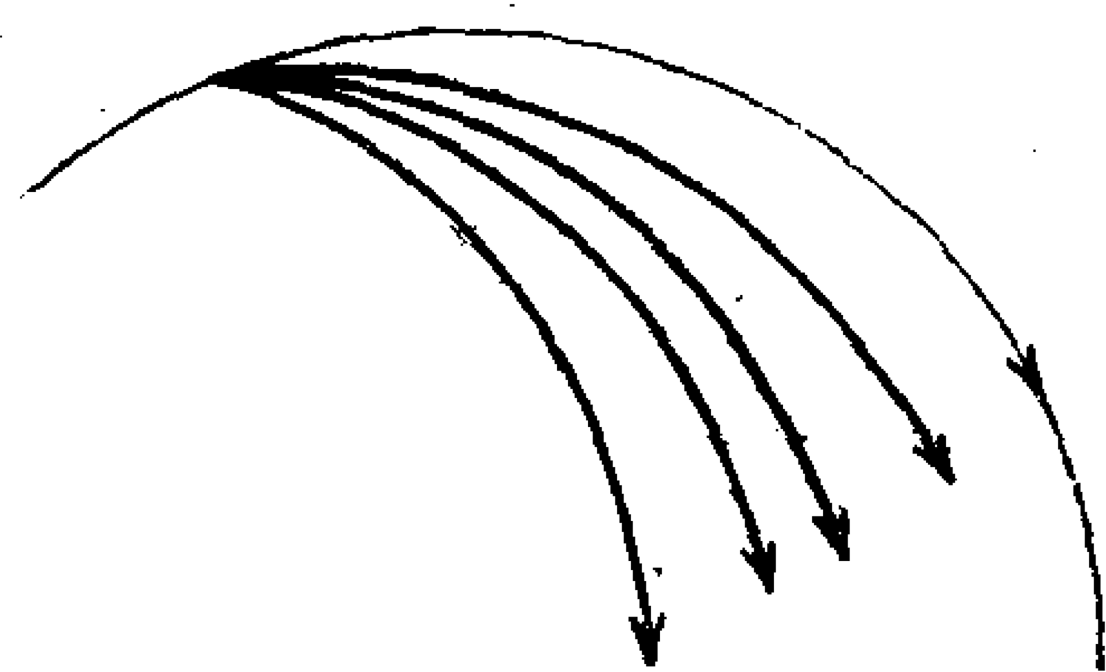


图 7-2-4

对于真实的地球来说，其速度分布总是符合$\frac{dV}{dr}<0$，即随深度而增加的。因此，总能观测到地震波，绘出走时曲线。由走时曲线求得的射线参数的$p$值随$\Delta$增大而不断减小。

由实测走时曲线研究表明，尽管地球内部介质波速随深度增加而变大，但仍然存在许多速度异常带及速度间断面。下面讨论这些异常带及间断面对射线及走时的影响。

## (二) 低速层及低速间断面情况

如图 7-2-5 所示，低速层是指满足下列速度分布关系的层，

$$\frac{dV(r)}{dr}<\frac{V(r)}{r}, \qquad 当\quad R\geqslant r>r_1,\ r<r_2;$$

$$\frac{dV(r)}{dr}>\frac{V(r)}{r}, \qquad 当 r_1>r>r_2. \tag{7-2.3}$$

这意味着在$r_1>r>r_2$层内，速度随$r$减小而下降得较快，进入该层的射线不满足(7-2.1)式，故折向地心。如图7-2-5(a)示，射线$EA$的最低点为$M$，$r_M=r_1$，对于入射角更小的射线，则由于低速层的影响而向下转折。直到射线穿过深度$r=r_2$，到达$r=r_3$处才出现最低点，然后向上折回到地面的$B$点。$r_3$满足关系

$$\frac{V(r_3)}{r_3}=\frac{V(r_1)}{r_1}. \tag{7-2.4}$$

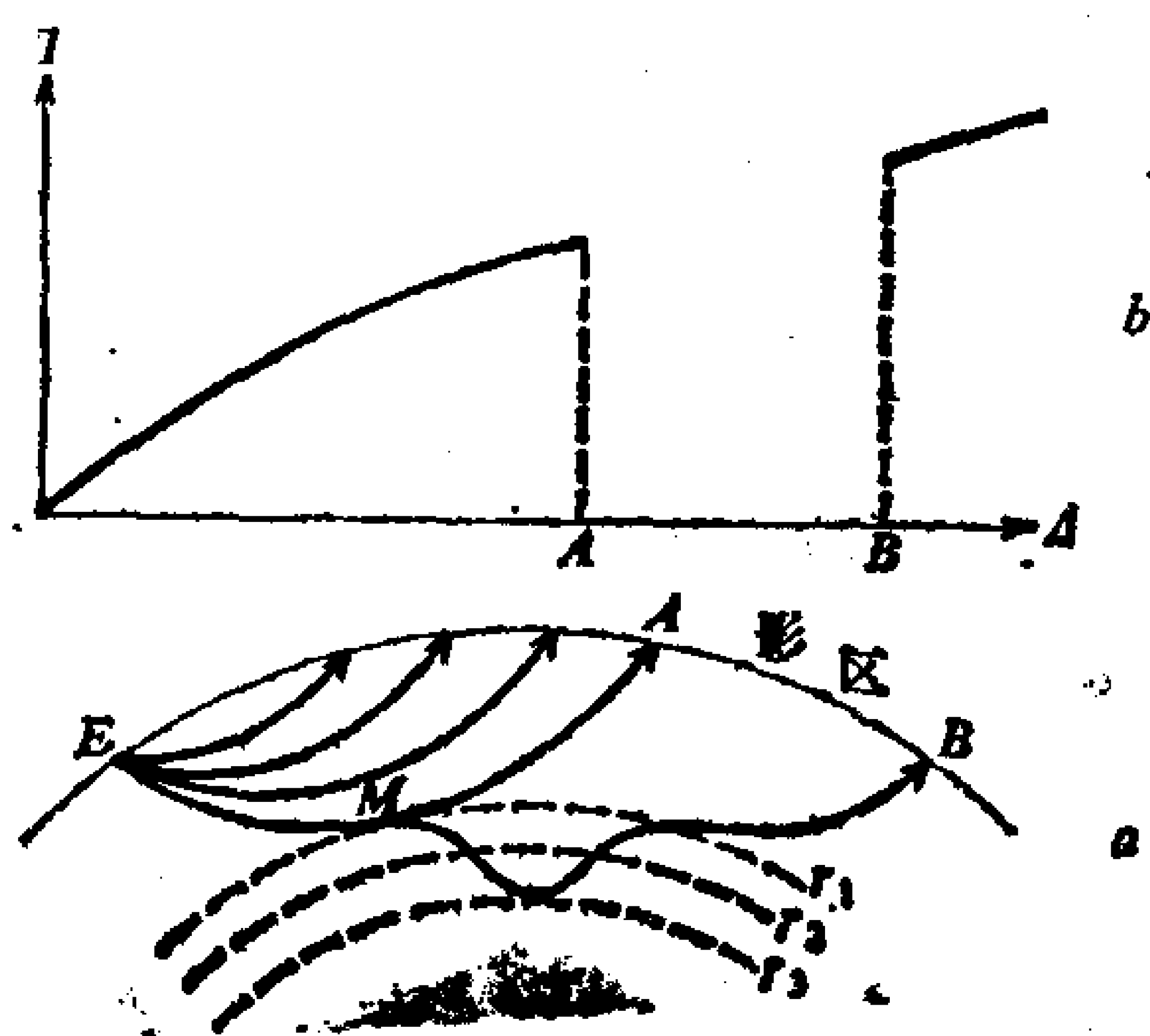



图 7-2-5

这时在地面上$AB$段便接收不到地震波而成为影区。对应的走时曲线上出现间断如图7-2-5(b)示。

当$r_1 \doteq r_2$，且上下都有$\frac{dv(r)}{dr}<0$（正的速度梯度），这个间断面称为低速间断面。这时，除直达波外，还将在界面上出现折射波及反射波，而$BC$区仍为影区。如图7-2-6示。

## （三）高速层和高速间断面

设在$r_1$到$r_2$的层中速度随深度的增加要比其上层及下层都快些，如图7-2-7示。速度变化关系满足下式

$$\left.\frac{dV(r)}{dr}\right|_{r_2<r<r_1}<\left.\frac{dV(r)}{dr}\right|_{\substack{r>r_1\\ r<r_2}}<0. \tag{7-2.5}$$

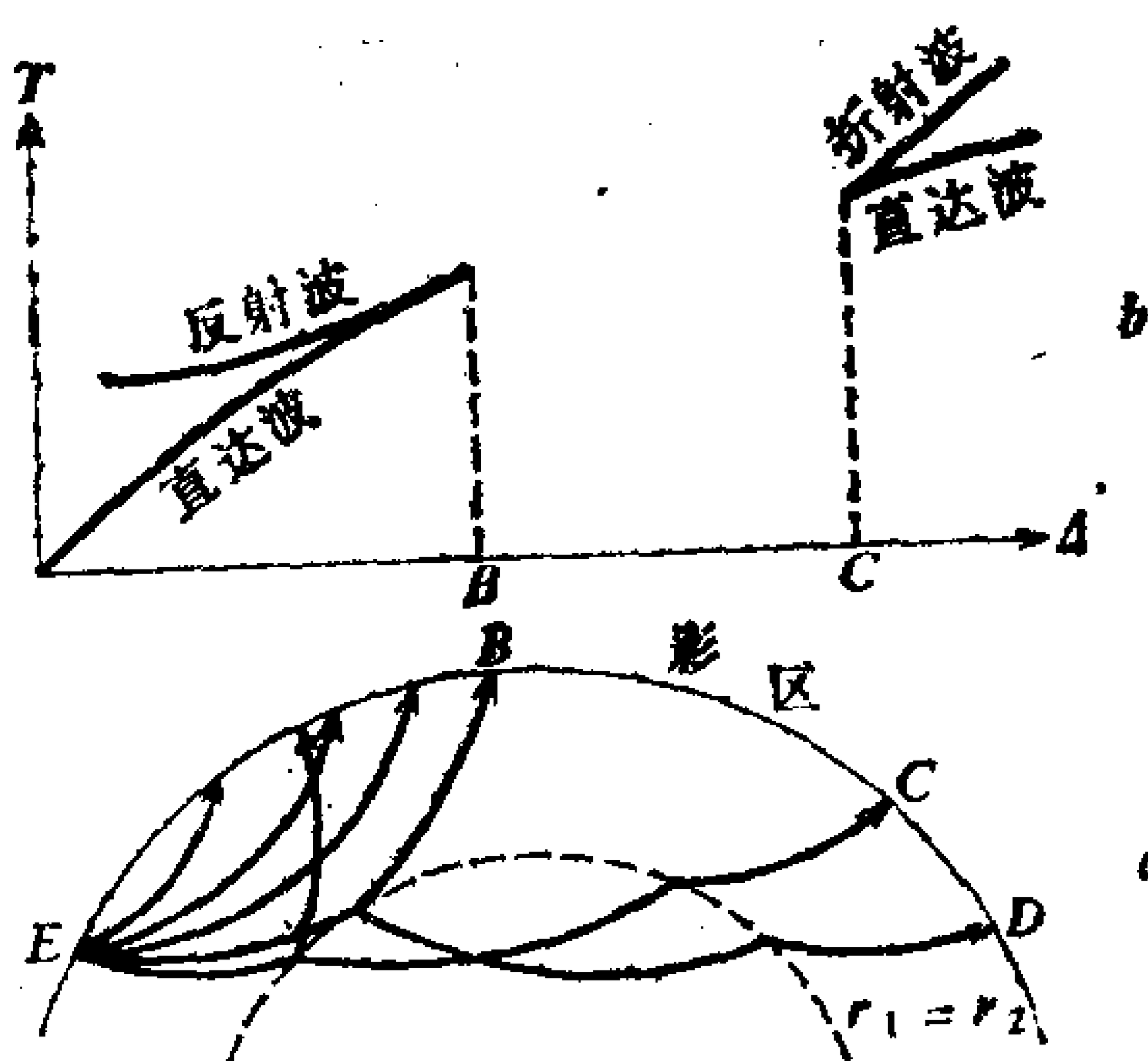



图 7-2-6

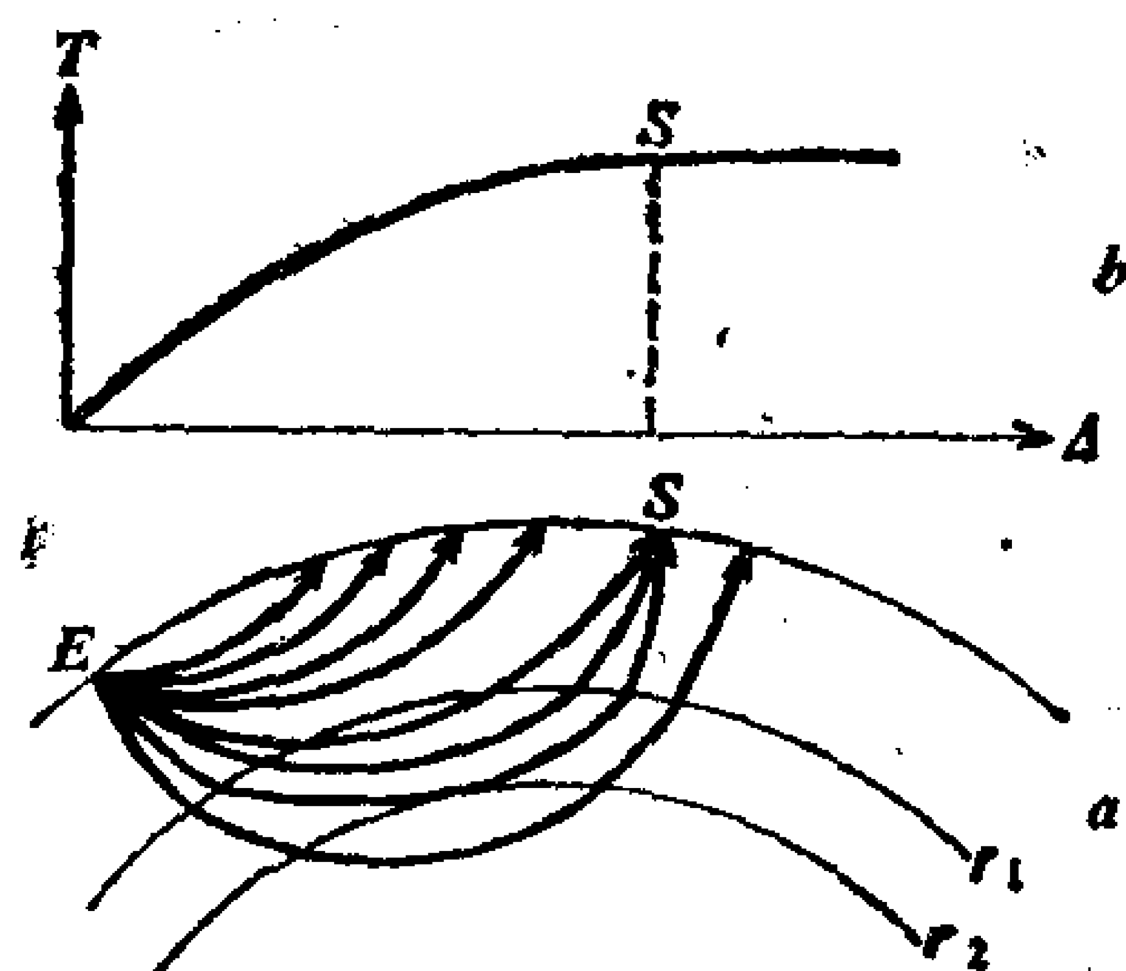



图 7-2-7

因而，在高速层$r_1$至$r_2$之间的射线曲率较大。这样，会使射线到达地面发生密集的现象，甚至汇集到同一点出射，如图7-2-7(a)示。汇集处对应的走时曲线曲率突然增大以至出现角点。如图7-2-7(b)示。

当$r_1$与$r_2$之间速度随深度增大增加得更快时（即有更高的速度梯度），会发生射线的交叉如图 7-2-8(a) 示。即入射到更深处的射线返回地面出射点的震中距反而变小。这时，走时曲线出现逆进现象。如图7-2-8(b)中的$AB$段。

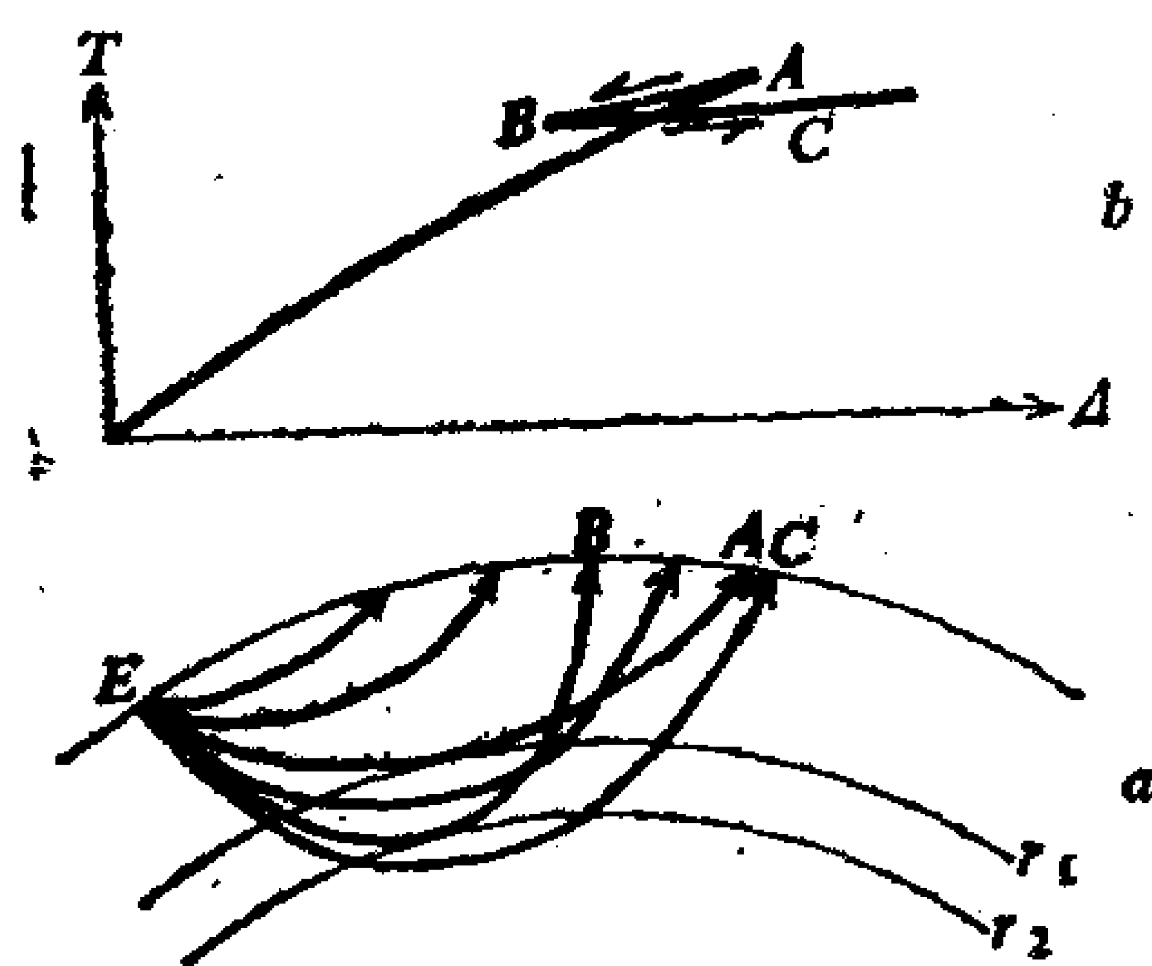



图 7-2-8

高速间断面是指在间断面下层的波速大于上层的波速，在间断面处波速$V$有一突增。现在仍讨论间断面的上方和下方均有$\frac{dV(r)}{dr}<0$的情况。射线及相应的走时曲线如图 7-2-9 所示。这时，波的回折和逆进的情况更为复杂如图7-2-9(a)。图7-2-9(b)中$OA$段为直达波，$GA$段相当于

反射波，BD段相当于折射波，B点相当于临界点。

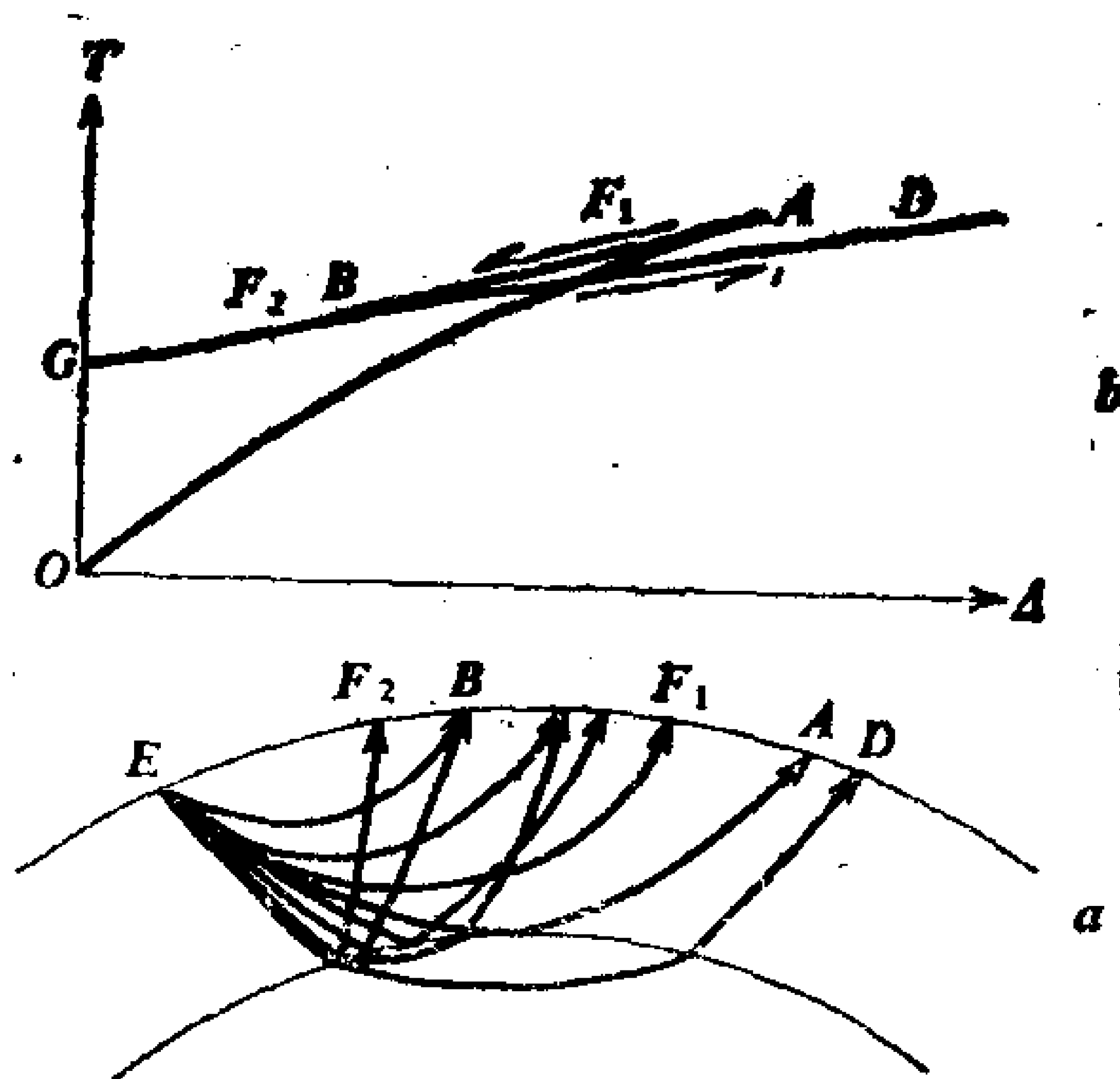



图 7-2-9

## §7-3 地球内部速度的反演

用体波走时曲线反演地球内部速度分布的传统方法是Herglotz-Wiechert方法。这是考虑一个球对称地球模型，速度$V$仅是地球半径$r$的函数，$V(r)/r$随着深度增加（半径$r$减小）而增大。在满足上述条件时，可以用此方法进行计算。

由(7-1.6)及(7-1.17)式可以分别重写为

$$\left.\begin{aligned} p &= \frac{r_M}{V(r_M)}, \\ p &= \frac{dT}{d\Delta}. \end{aligned}\right\} \quad (7\text{-}3.1)$$

式中$r_M$是参数为$p$的射线入射到地下最深处的地球半径。另一方面，$p$又等于观测走时曲线上该射线地面出射点处的斜率，即射线参数$p$可以由走时曲线上求得。由上式不难看出，如果能求出某一条射线穿透最深点的半径$r_M$，则速度$V(r_M)$也随之计算出来。对走时曲线上不同的距离$\Delta$进行计算，就可以得到地球内部

不同深度的速度。

(7=1.14)中的第一式中的被积函数及积分限均与$r_M$有关，可以通过它来计算$r_M$。置换变量

$$\eta=\frac{r}{V(r)},$$

于是(7-1-14)第一式写为

$$\Delta=2\int_{\eta_M}^{\eta_0}\frac{p}{\sqrt{\eta^2-p^2}}\frac{d\ln r}{d\eta}d\eta. \qquad (7\text{-}3.2)$$

其中 $\eta_0=\frac{R}{V_0}$，$\eta_M=\frac{r_M}{V_M}=p$.

$R$为地球平均半径，$V_0$为地球表面波速。

对(7-3-2)式两边均乘以，

$$\int_{\xi}^{\eta_0}\frac{dp}{\sqrt{p^2-\xi^2}},$$

得

$$\int_{\xi}^{\eta_0}\frac{\Delta dp}{\sqrt{p^2-\xi^2}}=2\int_{\xi}^{\eta_0}\frac{pdp}{\sqrt{p^2-\xi^2}}\int_{p}^{\eta_0}\frac{\frac{d\ln r}{d\eta}}{\sqrt{\eta^2-p^2}}d\eta.$$

其中$\xi=r_1/V(r_1)$，$r_1$为任意半径。上式右端一致收敛。由图7-3-1，改变二重积分秩序，上式可化为

$$\int_{\xi}^{\eta_0}\frac{\Delta dp}{\sqrt{p^2-\xi^2}}=2\int_{\xi}^{\eta_0}\frac{d\ln r}{d\eta}\int_{\xi}^{\eta}\frac{pdp}{\sqrt{(\eta^2-p^2)(p^2-\xi^2)}}. \qquad (7\text{-}3.3)$$

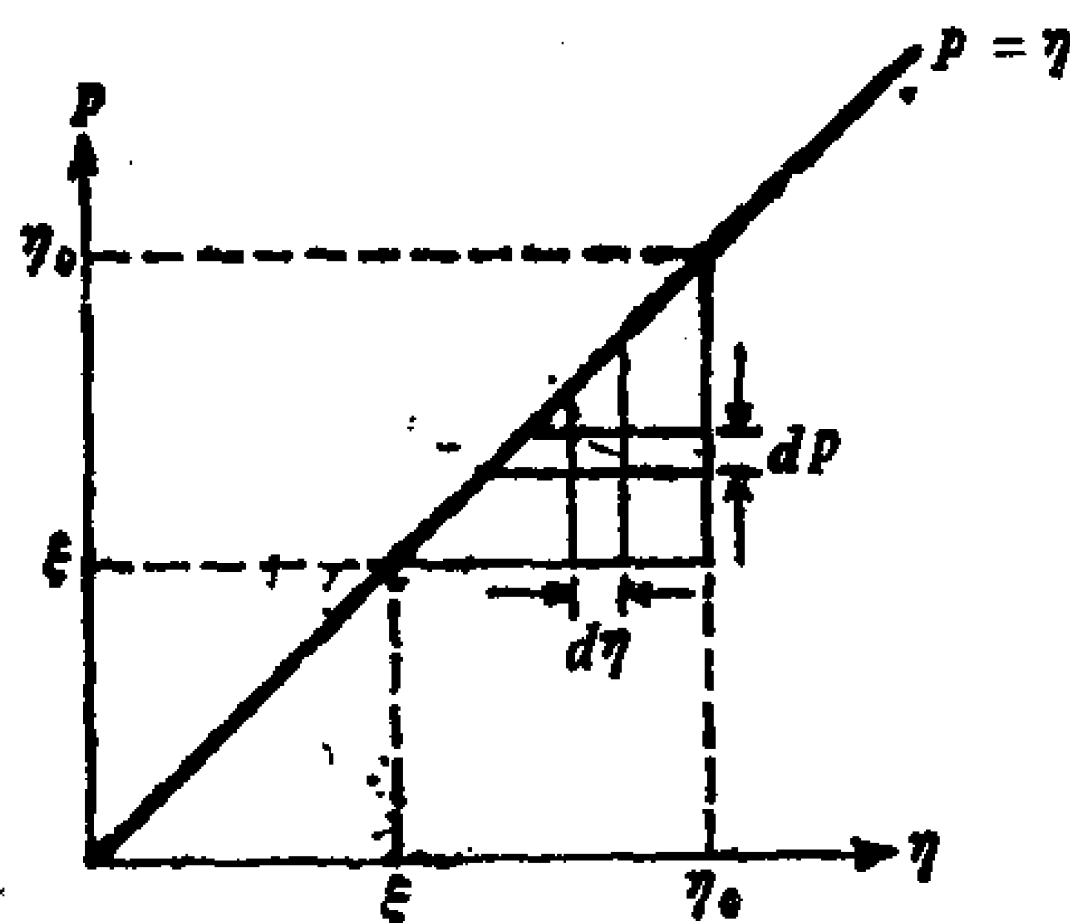


图 7-3-1

当$\eta > \xi$时，恒有关系

$$\int_{\xi}^{\eta} \frac{p d p}{\sqrt{(\eta^2 - p^2)(p^2 - \xi^2)}} = \frac{\pi}{2} .$$

所以(7-3.3)式可以简化为

$$\int_{\xi}^{\eta_0} \frac{\Delta d p}{\sqrt{p^2 - \xi^2}} = \pi \int_{\xi}^{\eta_0} \frac{d \ln r}{d \eta} d \eta$$

$$= \pi \int_{r_1}^{R} d \ln r = \pi \ln \frac{R}{r_1} . \qquad (7\text{-}3.4)$$

上式左端的积分为

$$\int_{\xi}^{\eta_0} \frac{\Delta d p}{\sqrt{p^2 - \xi^2}} = \int_{\xi}^{\eta_0} \frac{\Delta d(p/\xi)}{\sqrt{(p/\xi)^2 - 1}}$$

$$= \int_{p=\xi}^{p=\eta_0} \Delta d \left(\mathrm{ch}^{-1} \frac{p}{\xi}\right)$$

$$= \Delta \, \mathrm{ch}^{-1} \frac{p}{\xi} \Bigg|_{p=\xi}^{p=\eta_0} - \int_{\Delta_1}^{0} \mathrm{ch}^{-1}\left(\frac{p}{\xi}\right) d\Delta .$$

上式中 $p = \eta_0$时，射线最低点的半径为 $R$，即位于地表，这相当于$\Delta = 0$的情况。而当$p = \xi$时，$\mathrm{ch}^{-1}\left(\frac{\xi}{\xi}\right) = 0$，故上式右端第一项为零。由此(7-3.4)式可写为

$$\ln \frac{R}{r_1} = \frac{1}{\pi} \int_{0}^{\Delta_1} \mathrm{ch}^{-1} \frac{p}{\xi} d\Delta ,$$

或者写为

$$\ln \frac{R}{r_1} = \frac{1}{\pi} \int_{0}^{\Delta_1} \mathrm{ch}^{-1} \frac{p(\Delta)}{p(\Delta_1)} d\Delta . \qquad (7\text{-}3.5)$$

这就是所谓 Herglotz-Wiechert 公式。由式可见，只要由观测的走时曲线用数值微分法求出相应的射线参数曲线 $p(\Delta)$。则对于任意距离 $\Delta_1$，用(7-3.5)式右端计算积分值，从而求得其

对应的射线穿透地球内部的最深点的半径 $r_1$，以及该点的速度 $V_1$。这样便获得地球内部速度分布情况。

## §7-4　速度反演的具体计算方法

下面讨论用 Herglotz-Wiechert 公式进行数值反演的实际计算步骤。

### （一）实测走时曲线的校正

用来研究地壳上地幔速度分布的走时曲线往往是综合了许多地震到一系列选定的台站的到时。这些地震可能发生在大陆，也可能发生在海洋，而且具有不同的位置和深度。每个接收台站也具有不同的高程及位置。为了将这些走时及震中距归化到同一参考面上，必须作台站高度校正，地球椭圆扁率校正，震源深度校正等。如果只研究上地幔的速度分布，尚须作剥壳校正。

台站高程校正与地球的椭圆扁率校正公式与第五章§5-3所叙述完全相同。下面只讨论震源深度校正及剥壳校正。

震源深度改正是将不同震源深度的地震划归到同一深度面，它包括走时校正及震中距校正两项。

设校正的基准面深度为 $H_0$，震源深度为 $h_e$，可分别两种情况校正。

（1）$h_e<H_0$

震中距校正项

$$\delta\Delta_{h_e}=\frac{(H_0-h_e)\dfrac{V_1}{111.195}\dfrac{\partial t}{\partial\Delta}}{\sqrt{1-\left[\dfrac{V_1}{111.195}\dfrac{\partial t}{\partial\Delta}\right]^2}}\,,\qquad(7\text{-}4.1)$$

走时校正项

$$\delta t_{h_e}=\frac{H_0-h_e}{V_1\sqrt{1-\left[\dfrac{V_1}{111.195}\dfrac{\partial t}{\partial\Delta}\right]^2}}\,.\qquad(7\text{-}4.2)$$

（2） $h_e \geqslant H_0$时

震中距校正项

$$\delta\Delta_{h_e} = -\frac{(h_e - H_0)\dfrac{V_2}{111.195}\dfrac{\partial t}{\partial \Delta}}{\sqrt{1-\left[\dfrac{V_2}{111.195}\dfrac{\partial t}{\partial \Delta}\right]^2}}, \tag{7-4.3}$$

走时校正项

$$\delta t_{h_e} = -\frac{h_e - H_0}{V_2\sqrt{1-\left[\dfrac{V_2}{111.195}\dfrac{\partial t}{\partial \Delta}\right]^2}}. \tag{7-4.4}$$

上述各式中：$V_1$——校正基准面以上纵波平均速度，对于大陆区$V_1$可取6.5公里/秒，海洋区可取8.0公里/秒；$V_2$——校正基准面以下纵波平均速度，对于大陆区$V_1$可取7.9公里/秒，海洋区可取8.2公里/秒；$\dfrac{\partial t}{\partial \Delta}$——走时曲线的斜率（或慢度），可从走时曲线上获得。

如果震源深度校正的基准面不在地面，而在地下某一深度$H_0$（$H_0>0$），则地面的台站接收端的震中距及走时亦应校正到此基准面上，称为剥壳校正。其震中距校正项为

$$\delta\Delta_S = \frac{H_0\dfrac{V_1}{111.195}\dfrac{\partial t}{\partial \Delta}}{\sqrt{1-\left(\dfrac{V_1}{111.195}\dfrac{\partial t}{\partial \Delta}\right)^2}}, \tag{7-4.5}$$

走时校正项为

$$\delta t_S = \frac{H_0}{V_1\sqrt{1-\left(\dfrac{V_1}{111.195}\dfrac{\partial t}{\partial \Delta}\right)^2}}. \tag{7-4.6}$$

设第$j$个台站记到第$i$个地震的$p$波走时为$t_{ij}$，相应的震中距为$\Delta_{ij}$，则校正后的走时及震中距为

$$t'_{ij} = t_{ij} - (\delta t_{Hs} + \delta t_{he} + \delta t_e + \delta t_s),$$

$$\Delta'_{ij} = \Delta_{ij} - (\delta \Delta_{he} + \delta \Delta_s). \tag{7-4.7}$$

式中$\delta t_{Hs}$为台站高程校正项，$\delta t_{he}$、$\delta \Delta_{he}$为震源深度校正项，$\delta t_e$为地球椭圆扁率改正项，$\delta t_s$、$\delta \Delta_s$为剥壳校正项。

### （二）由走时曲线计算射线参数p

对经过校正的走时曲线$T$-$\Delta$用数值微分方法求$p = dT/d\Delta$，以获得一条连续的射线参数$p$-$\Delta$曲线。

在计算机上利用三次自然样条函数对一元函数进行成组插值及微分，具有较好的平滑性能，能保证所插值的函数及其一阶导数、二阶导数连续。计算方法如下。

设走时曲线上$n$个插值结点$x_1 < x_2 < \cdots < x_n$对应的函数值为$y_1$，$y_2$，…，$y_n$。

三次自然样条函数$S(x)$满足以下条件：①$S(x_i) = y_i, i = 1, 2, \cdots, n$；②在每个子区间$(x_i, x_{i+1})$上，$i = 1, 2, \cdots, n-1$，样条函数是一个三次多项式；③在$[x_1, x_n]$上有一阶及二阶连续导数，且在端点处有$S''(x_1) = S''(x_n) = 0$。下面分别给出$S(x)$，$S'(x)$及$S''(x)$的计算公式。

由牛顿插值定理可写出以下关系：

$$\begin{aligned} S(x) = & S(x_i) + (x - x_i) S(x_i, x_{i+1}) \\ & + (x - x_i)(x - x_{i+1}) S(x, x_i, x_{i+1}), \\ & X \in (x_i, x_{i+1}). \end{aligned} \tag{7-4.8}$$

式中 $$S(x_i, x_{i+1}) = \frac{y_{i+1} - y_i}{x_{i+1} - x_i},$$

$$S(x, x_i, x_{i+1}) = \frac{1}{6}[S''(x_i) + S''(x) + S''(x_{i+1})].$$

对(7-4.8)式求微分

$$S'(x)=S(x_i,\ x_{i+1})+(2x-x_i-x_{i+1})S(x,\ x_i,\ x_{i+1})$$
$$+\frac{1}{6}(x-x_i)(x-x_{i+1})S''(x_i,\ x_{i+1}),$$
$$X\in(x_i,\ x_{i+1}).$$
(7-4.9)

由于$S''(x)$在$(x_i,\ x_{i+1})$区间是直线，因此有

$$S''(x)=S''(x_i)+(x-x_i)S''(x_i,\ x_{i+1}).$$
(7-4.10)

根据在交接点一阶导数连续的条件，可以写出下述$n-1$个方程

$$(x_i-x_{i-1})S''(x_{i-1})+2(x_{i+1}-x_{i-1})S''(x_i)$$
$$+(x_{i+1}-x_i)S''(x_{i+1})=6[S(x_i,\ x_{i+1})-S(x_{i-1},\ x_i)],$$
$$i=2,\ 3,\ \cdots\cdots,\ n-1。$$
(7-4.11)

给定边界条件$S''(x_1)=S''(x_n)=0$及$S'(x_1)=S'_1, S'(x_n)=S'_n$。

待解方程组(7-4.11)可以写为下列矩阵形式

$$\begin{pmatrix} b_0 & c_0 & & & & 0 \\ a_1 & b_1 & c_1 & & & \\ & a_2 & b_2 & c_2 & & \\ & & \ddots & \ddots & \ddots & \\ 0 & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & a_n & b_n \end{pmatrix}\begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{pmatrix}=\begin{pmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \quad (7\text{-}4.12)$$

式中各元素为

$$a_i=x_i-x_{i-1},$$
$$b_i=2(x_{i+1}-x_{i-1}),$$
$$c_i=x_{i+1}-x_i,$$
$$d_i=6\left(\frac{y_{i+1}-y_i}{x_{i+1}-x_i}-\frac{y_i-y_{i-1}}{x_i-x_{i-1}}\right).$$

未知数$u_i$为待求的二阶导数$S''(x_i)$值。

(7-4.12)式是一个对角元占优势的三对角线带状矩阵，可以用追赶法求解。求得在结点上的二阶导数值$S''(x_i)$。

然后，分别利用(7-4.8)，(7-4.9)及(7-4.10)式，即可求得走时曲线上每个插值点$t_j(j=1, 2, \cdots, m)$上的函数值$S(t_j)$及一阶导数值$S'(t_j)$。将$S'(t_j)$值连接起来，就可以获得射线参数$p$的光滑曲线。

在处理实际观测资料时，考虑到观测误差的存在，故对实测数据进行样条函数平滑。这样，可以获得更合理的射线参数$p$值曲线。其算法原理如下：

设给定$n$个点$x_1<x_2<\cdots<x_n$，其对应的测定值为$y_1$，$y_2$，$\cdots y_n$。我们要构造平滑函数$g(x)$，使

$$\int_{x_1}^{x_n}[g''(x)]^2dx=\min,$$

且

$$\sum_{i=1}^{n}\left(\frac{g(x_i)-y_i}{\delta y_i}\right)^2\leqslant S,$$

其中$\delta y_i(i=1, 2, \cdots, n)$是给定的，一般可以取为对$y_i$的标准偏差估计值。$S$也是给定的，一般常令$S=n$。当$S=0$时，表示计算样条函数插值。利用变分法即可解出$a_i$，$b_i$，$c_i$及$d_i$的值，从而得到光滑的$p$值曲线。在平滑过程中，可以选择不同的$\delta y_i$值，以取得最佳的计算结果。

## （三）　根据射线参数$P$求速度分布

对(7-3.5)式右端积分式

$$\int_0^{\Delta_1}\mathrm{ch}^{-1}(p/p_1)d\Delta$$

中的被积函数改写为下列公式

$$\mathrm{ch}^{-1}(X) = \pm \ln(X + \sqrt{X^2+1}). \quad (X \geqslant 1)$$

因此积分式即可以写为

$$\int_0^{\Delta_1} \ln(p/p_1 + \sqrt{(p/p_1)^2 - 1}) d\Delta . \qquad (7\text{-}4.13)$$

只要地球内部速度分布满足 $V(r)/r$ 比 值随深度递增时，则总有$p/p_1 \geqslant 1$，积分(7-4.13)式总可以算出。

选定一系列的震中距 $\Delta_1$ 值，用数值积分(例如，辛普生积分公式)可以得到一系列的积分值

$$S_1 = \int_0^{\Delta_1} \ln(p/p_1 + \sqrt{(p/p_1)^2 - 1}) d\Delta. \qquad (7\text{-}4.14)$$

由(7-3.6)式有

$$S_1 = \pi \ln(R/r_1),$$

从而求得相应于$\Delta_1$距离处射线穿透地球最深点的半径

$$r_1 = R/\exp(S_1/\pi). \qquad (7\text{-}4.15)$$

根据(7-1.6)式，可求得该点的速度为

$$V_1 = \frac{r_1}{p_1} . \qquad (7\text{-}4.16)$$

再由半径 $r_1$换算为深度 $H_1 = R - r_1$，就得到地球内部速度 $V_1$随深度$H_1$的变化。

最后，讨论一下近距离走时曲线的速度反演问题。由于这时走时-距离关系用直角座标表 示。Herglotz-Wiechert 公式(7-3.5)可以写成

$$Z(V_P) = \frac{1}{\pi}\int_0^{x_P} \mathrm{ch}^{-1} \frac{V(x_P)}{V(x)} dx ,$$

式中 $x_P$表示地面任一 $P$ 点的震中距，$Z(V_P)$ 表示相应于视速度为$V_P$的射线穿透的最深点，$V(x)$为沿$x$方向视速度变化值$dx/dT$。只要由走时曲线求得视速度光滑曲线，就可以反演地下介质速度分布。

## §7-5 反演方法的改进

上一节讨论的 Herglotz-Wiechert 速度反演计算方法，只适用于速度分布 $V(r)/r$ 随深度增大的情况。但是，由于地球内部低速层的存在，往往不满足上述条件，故该方法在使用时受到一定限制。

1966年，苏联 Gerver和 Markushevich 提出了一种计算方法，它可以对低速层进行反演。1977年，美国McMechan等又将反演方法作了简化。由于这些计算都是基于平坦地球，因此必须作坐标变换。

在球形空间，我们用的坐标变量为$(r, \theta)$。在半无限空间中为$(x, y)$，其中$x$为平坦地球的横坐标，$y(\geqslant 0)$为深度(纵向)坐标。如图 7-5-1 示。利用保角变换，将圆映射到半无限空间，其变换公式为

$$x=\frac{R}{V(R)}\theta,$$
$$y=\frac{R}{V(R)}\ln\frac{R}{r}\quad. \tag{7-5.1}$$

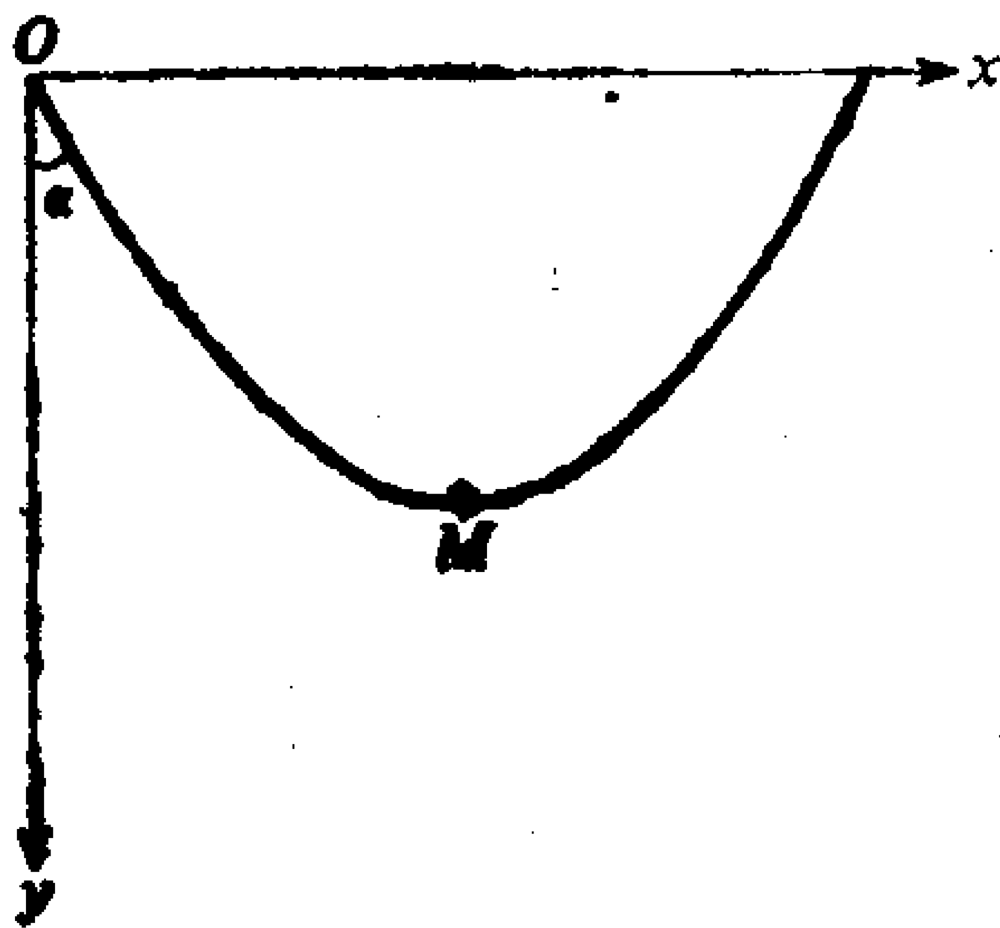



图 7-5-1

式中 $R$ 为地球平均半径（6370 公里）。由公式显见，在地面上$(r=R)y=0$。

在此坐标系中导出速度$V$的变换值为

$$u(y)=\frac{V\left(Re^{-\frac{V(R)}{R}y}\right)}{V(R)e^{-\frac{V(R)}{R}y}}\quad. \tag{7-5.2}$$

变换后的速度$u$仅是深度$y$的函数。它在地面$u(0)=1.0$，向下逐渐增大($u>1.0$)。

射线参数的定义为

$$p=\frac{\sin\alpha(y)}{u(y)}, \qquad p\in(0,\ 1) \tag{7-5.3}$$

其中$\alpha$角为射线与$y$轴的夹角，$0<\alpha<90°$。图7-5-1中绘出某一条穿透最深点$M$回折到地面的射线。

由图7-5-2可见，设某一条射线参量为$p$的射线向下入射，则有

$$\sin\alpha(y)=pu(y). \tag{7-5.4}$$

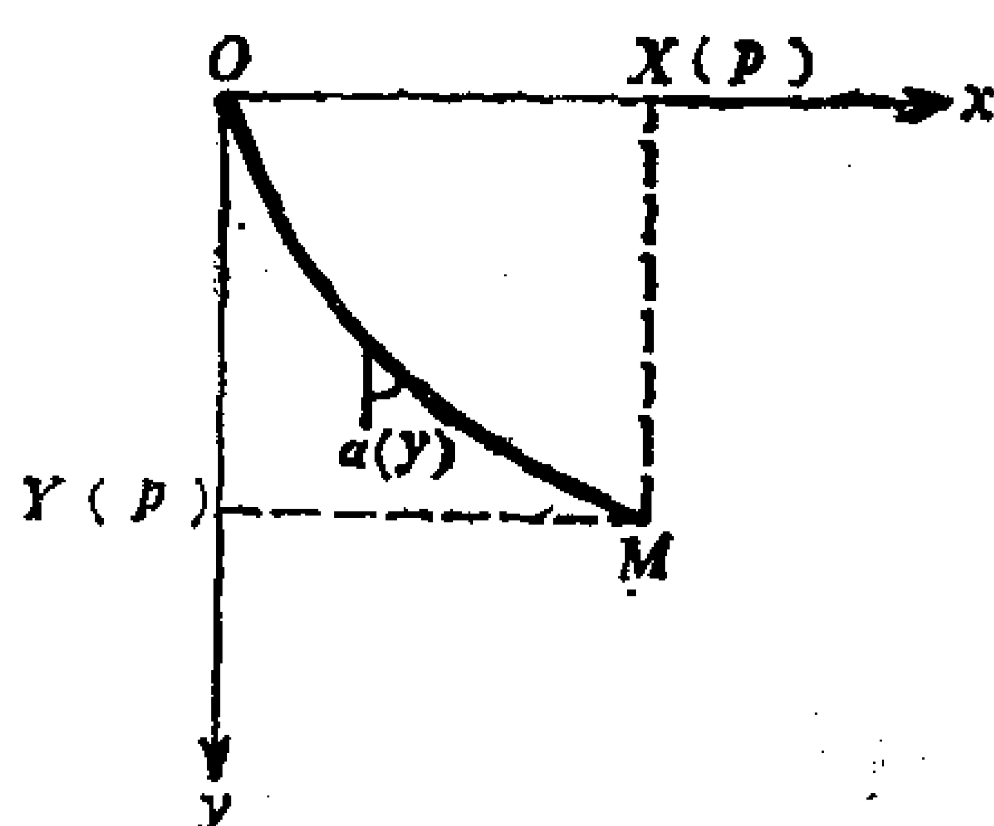



图 7-5-2

且

$$\frac{dx}{dy}=\operatorname{tg}\alpha(y)=\frac{\sin\alpha(y)}{\cos\alpha(y)}=\frac{pu(y)}{\sqrt{1-p^2u^2(y)}},$$

于是得射线上任一点$x$与$y$的关系为

$$x=\int_0^y\frac{pu(y)dy}{\sqrt{1-p^2u^2(y)}}. \tag{7-5.5}$$

地震波从地面到达$(x,\ y)$点的走时为

$$t=\int_0^y\frac{dy}{u(y)\cos\alpha(y)}$$

$$=\int_0^y\frac{dy}{u(y)\sqrt{1-p^2u^2(y)}}. \tag{7-5.6}$$

由图7-5-2可见，对于射线参数为$p$的射线，其回折点$M$处的坐标$X(p)$及$Y(p)$，以及从地面到达该点的走时的关系为

$$X(p)=\int_0^{Y(p)}\frac{pu(y)dy}{\sqrt{1-p^2u^2(y)}},$$

$$T(p)=\int_0^{Y(p)}\frac{dy}{u(y)\sqrt{1-p^2u^2(y)}}. \tag{7-5.7}$$

以上就是由已知速度分布$u(y)$计算地面走时曲线$X(p)$及$T(p)$的公式，它和球对称地球计算$\Delta$及$T$的公式(7-1.14)是类似的。在正演公式(7-5.7)中，地下速度分布$u(y)$是任意的，即可以包含低速层。

现在的问题是求反演问题，即从观测得到的$X(p)$及$T(p)$去计算速度分布$u(y)$。为此，先引入一个辅助函数$\tau(p)$定义

$$\tau(p)=T(p)-pX(p). \tag{7-5.8}$$

因为

$$T(p)-pX(p)=\int_0^{Y(p)}\frac{dy}{u(y)\sqrt{1-p^2u^2(y)}}$$

$$-p\int_0^{Y(p)}\frac{pu(y)dy}{\sqrt{1-p^2u^2(y)}}=\int_0^{Y(p)}\frac{1-p^2u^2(y)}{u(y)\sqrt{1-p^2u^2(y)}}dy$$

$$=\int_0^{Y(p)}\sqrt{u^{-2}(y)-p^2}\,dy,$$

故有

$$\tau(p)=\int_0^{Y(p)}\sqrt{u^{-2}(y)-p^2}\,dy. \tag{7-5.9}$$

由$X(p)$及$T(p)$求速度分布问题，便化为由辅助函数$\tau(p)$根据(7-5.9)式求$u(y)$的问题。因为$\tau(p)$可由(7-5.8)式实际算出，所以问题便归结为求出射线参数为$p$的最深点的纵坐标$Y(p)$。



地
6—

从一组实测走时值$(T_i, x_i)$计算$\tau(p)$简单可靠的办法是基于$T(p_0)-pX(p_0)$在$p_0$点有极大值的原理。故每固定一个$p$值，绘$\tau=T_i-pX_i$曲线图，当此曲线取极大值时，所对应的$\tau$值即为所求的$\tau(p)$值。对一系列$p$值求$\tau(p)$，则可得一条$\tau(p)$曲线。

$Y(p)$的反演公式为

$$Y(p)=\frac{1}{\pi}\int_0^{\tau(p)}\frac{d\tau}{\sqrt{p(\tau)-p}}\ . \tag{7-5.9a}$$

式中$p(\tau)$与$\tau(p)$互为反函数，且$Y(p)=Y(c)|_{c=1/p}$

$\tau(p)$曲线的主要优点是能解开走时曲线的多重性。即当地下存在强速度梯度带时，走时曲线$T(x)$逆进出现三重区时，$\tau(p)$也是单调降函数；假若存在低速带，$\tau(p)$曲线也只有简单的跳跃间断。

Gerver和Markushevich推广了Herglotz-Weichert方法。当地下速度分布包含有低速层时（如图7-5-3），射线参数为$q$的射线最低点的深度应为

$$Y(q)=\frac{2}{\pi}\int_q^1\frac{X(p)dp}{\sqrt{p^2-q^2}}+\sum_{i=0}^{k}\frac{2}{\pi}\int_{y_i}^{\overline{y_i}}\mathrm{arctg}\sqrt{\frac{u^{-2}(y)-p_i^2}{p_i^2-q^2}}dy. \tag{7-5.10}$$

式中前一项为主项，它实际上是不存在低速层时的Herglotz-Wiechert反演计算公式。后一项称为低速层的附加项，即低速层对反演式的附属累加值。$y_i-\overline{y_i}$是第$i$个低速层的厚度。$p_i$是掠过第$i$个低速层顶部射线的参数。$u(y)$是速度分布；$K$是低速层的个数。

在应用这一公式反演时，必须对低速层的厚度$y_i-\overline{y_i}$及速度分布$u(y)$作出假定，以对实测走时曲线进行校正，然后进行反演。低速层参数假设不同，反演结果也不同。产生这种不唯一性的原因是低速层的存在，地面将出现阴影带，无法观测到与低速层

相应的走时曲线。

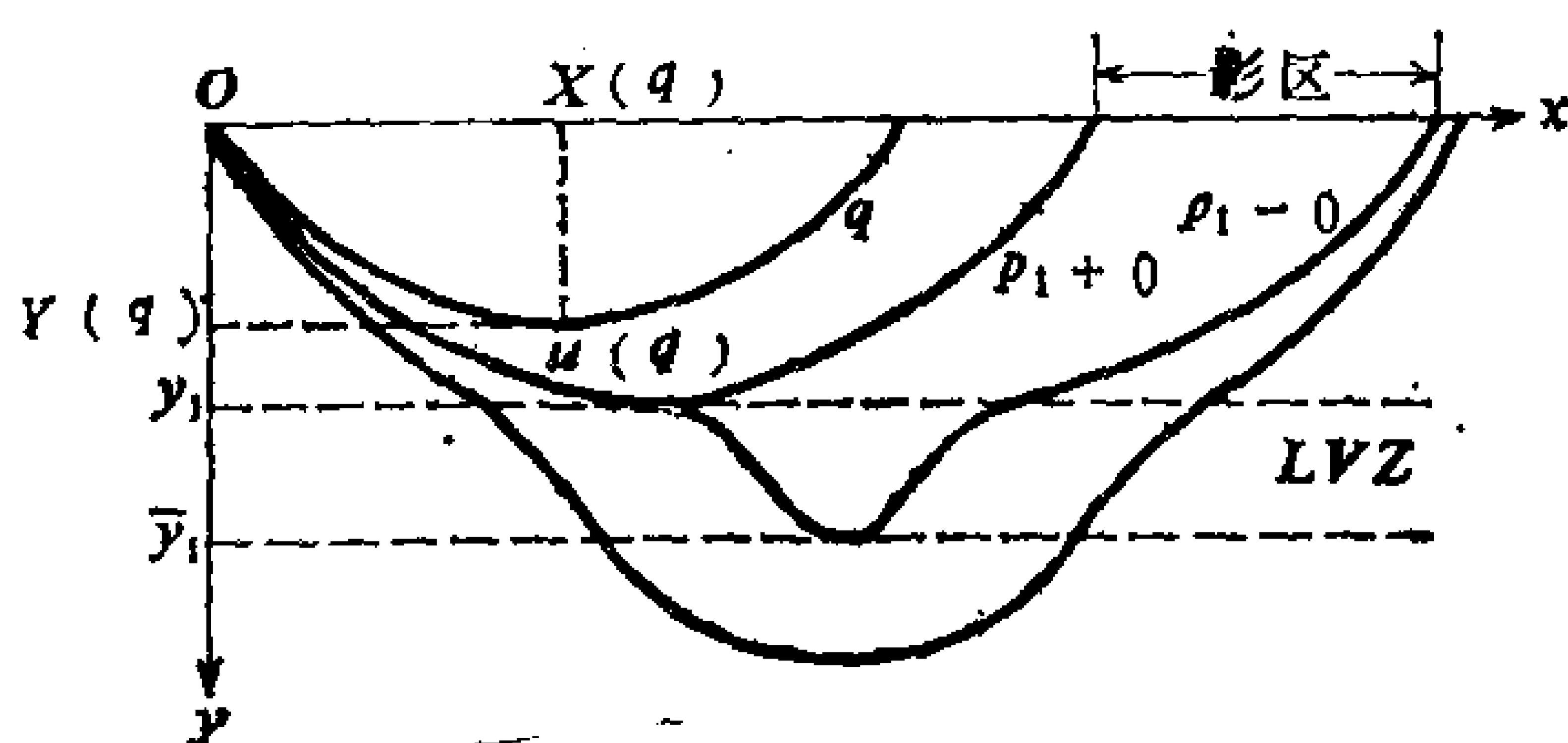



图 7-5-3

设射线的传播路径如图7-5-4(a) 所示，Ⅰ、Ⅲ 为高速层，Ⅱ 为低速层。设想把实际存在的低速层抽去，将$p<p_1$的射线的路径向左移动，如图7-5-4(b)所示。这样就使同一参数的射线在Ⅰ、Ⅲ介质中重新连接起来。

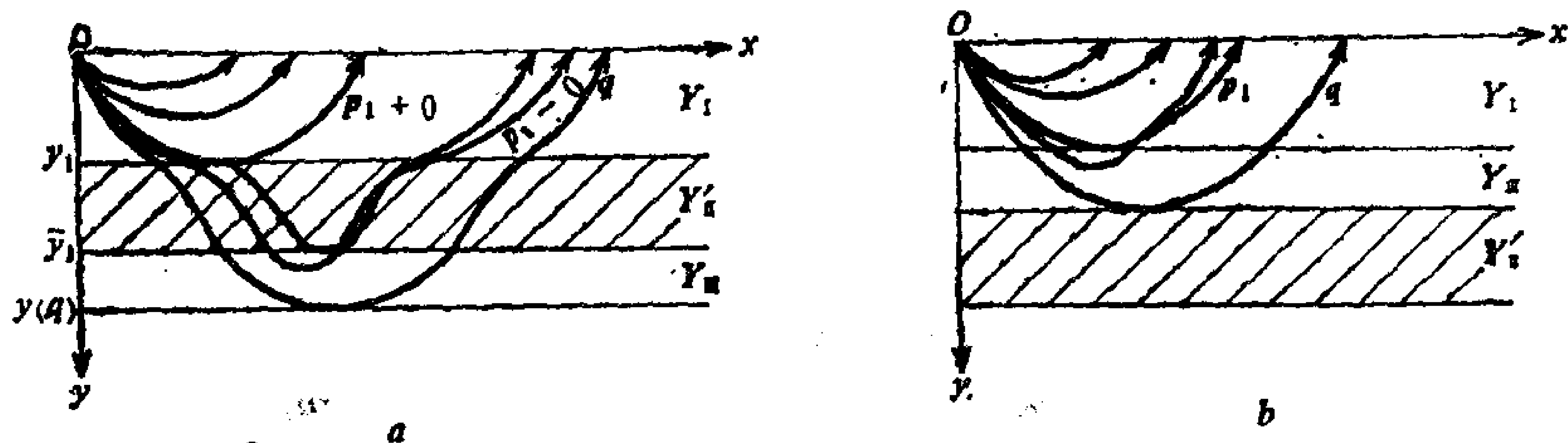



图 7-5-4

由图可见，抽去低速层连接起来的射线路径，与不存在低速层Ⅱ而仅有层Ⅰ、Ⅲ传播的路径完全相同。因此，穿过低速层参数为$q$的射线最低点的深度可改写为

$$Y(q)=Y_Ⅰ+Y_Ⅱ+Y_Ⅲ=(Y_Ⅰ+Y_Ⅲ)+Y_Ⅱ$$

$$=\frac{2}{\pi}\int_q^1\frac{X(p)dp}{(p^2-q^2)^{1/2}}+(\overline{y_1}-y_1)$$

$$(0\leqslant q<p_1) \qquad (7\text{-}5.11)$$

式中X(p)应是没有低速层时，射线仅在Ⅰ、Ⅲ层中传播的对应的慢度曲线。当射线在Ⅰ层中时，$p_1 \leqslant p \leqslant 1$，X(p)即实测的 p-X 曲线。但当射线穿透到低速层以下时，$0 \leqslant p < p_1$，X(p)与实测曲线已不符合。为了使射线联接成图7-5-4(b)的样子，就应将实测p-X曲线向左作相应移动，即计算

$$X(p) = X(p)_{实测} - S_1(p)/2, \tag{7-5.12}$$

式中

$$S_1(p) = 2\int_{y_1}^{\overline{y_1}} \frac{p}{\sqrt{u^{-2}(y) - p^2}} dy,$$

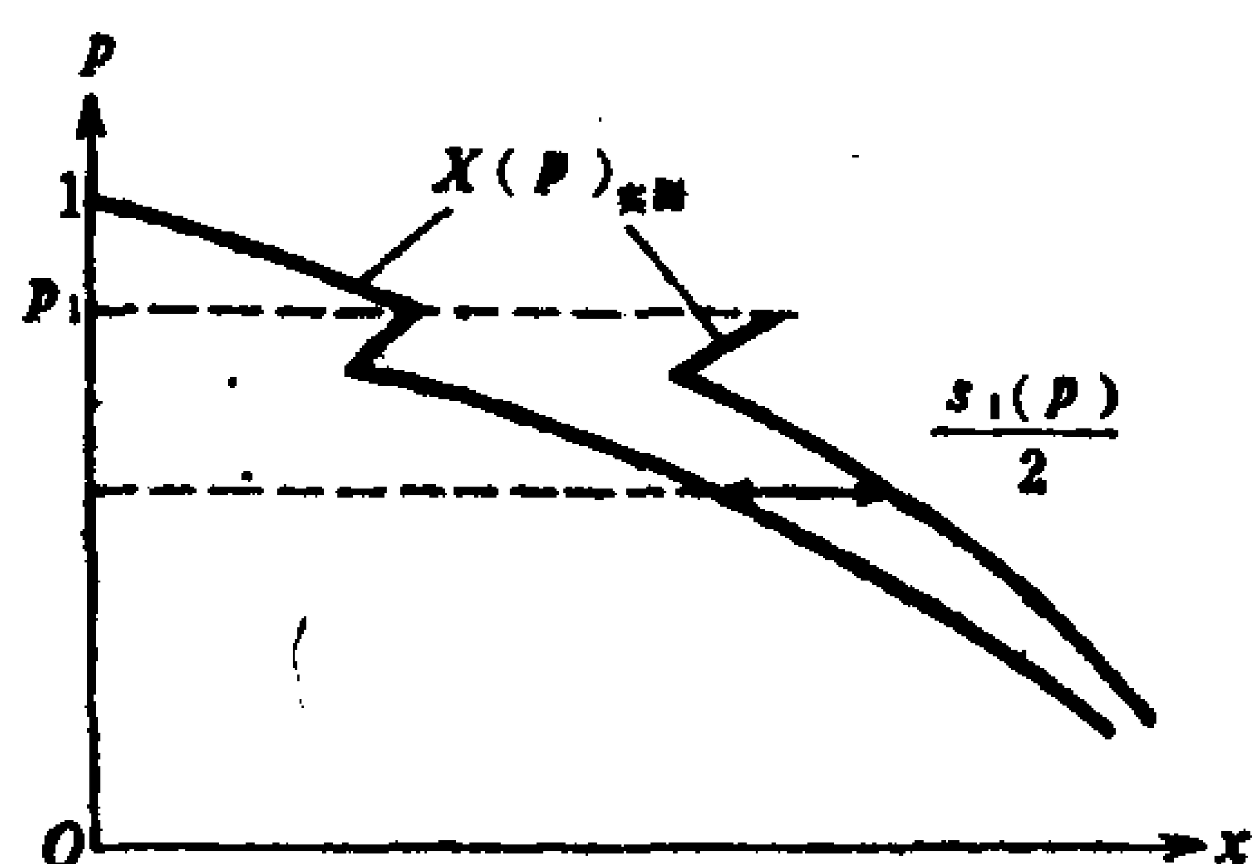



图 7-5-5

表示参数为p的射线在低速层中所经历的水平距离，校正如图7-5-5所示。

当存在多个低速层时，同理可得

$$Y(q) = \frac{2}{\pi}\int_q^1 \frac{X(p)dp}{\sqrt{p^2 - q^2}} + \sum_{i=1}^{k} (\overline{y_i} - y_i), \tag{7-5.13}$$

式中

$$X(p) = X(p)_{实测} - \sum_{i=1}^{k} S_i(p)/2$$

而

$$S_i(p) = 2\int_{y_i}^{\overline{y_i}} \frac{p}{\sqrt{u^{-2}(y) - p^2}} dy$$

注意，以上公式中参数p, u(y)，X与球座标中参数的关系为

$$X = \frac{x}{2} = \eta_0 \frac{\Delta}{2}$$

$$u(y) = \eta_0 \frac{V(r)}{r}$$

$$p = p_{球}/\eta_0$$

而 $\eta_0 = R_0/V_0$，$R_0$及$V_0$分别表示地球平均半径及地表平均速度。由上述换算关系就可以容易地求出地球内部速度分布。

式(7-5.10)表明，如果存在低速层，必须对各低速层的厚度$y_i - \overline{y_i}$作出假定，才能求得深度$Y(q)$。而这一点是难于做到的。下面我们讨论McMechan引入的一个改进方法。

McMechan指出，低速层的反演计算的主要难点在于该层中没有回折点，即没有从低速层中返回地面的射线。为此，他建议用分段反演方法解决这个问题。在每一个反演单元中，$p$值是单调变化的。

图7-5-6为两个$p$值单调变化的速度分层单元。其虚线表示各射线回折点的轨迹。(a)为正的速度梯度层，(b)为负的速度梯度层。

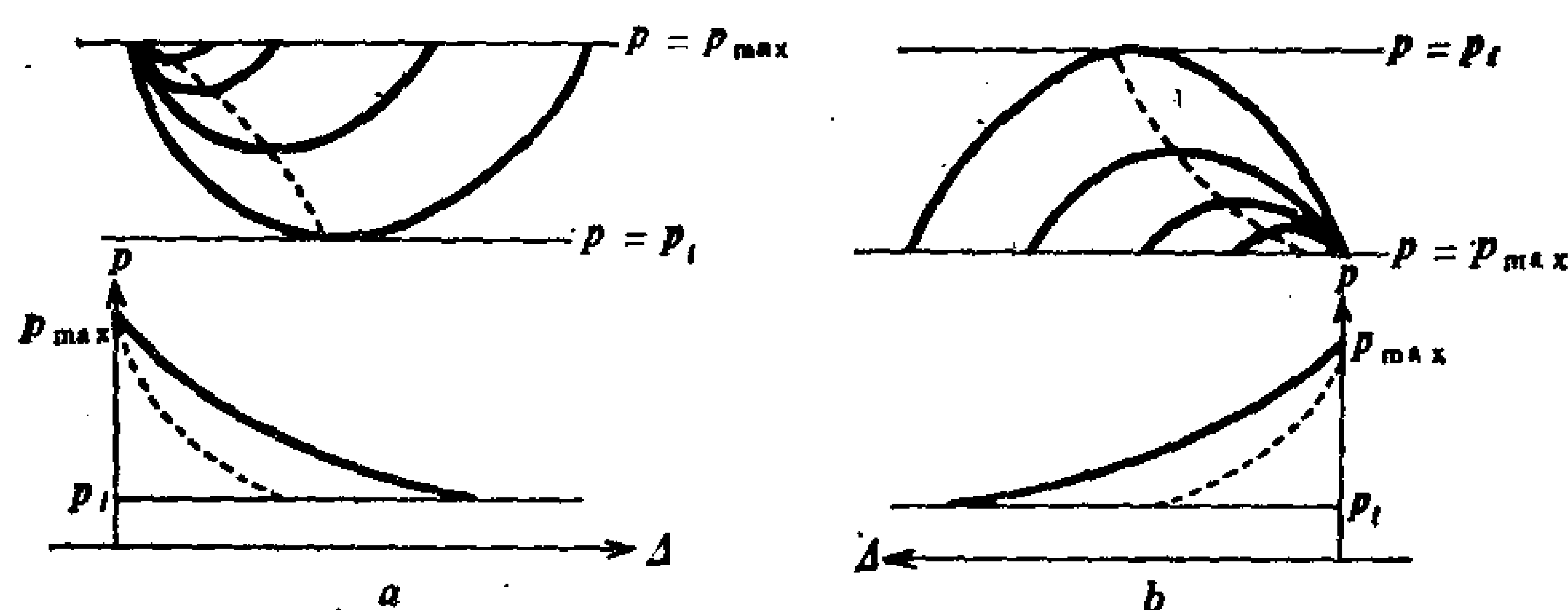



图 7-5-6

由这样划分的$p$值单调变化分层称为反演单元。在每一个单元中，从顶界面到下界面$p$值都是单调变化的。因此，可以分别对每一单元用Herglotz-Weichert公式反演，以得到速度随深

度的变化。引用(7-5.10)式中的主项，写出

$$Y(p_i)-Y(p_{max})=\frac{2}{\pi}\int_{p_i}^{p_{max}}\frac{X(p)dp}{(p^2-p_i^2)^{1/2}}. \qquad (7\text{-}5.14)$$

式中$Y(p_i)-Y(p_{max})$是从最大的$p$值(通常在地面)到$p_i$的深度增量。对于负的速度梯度层，深度增量应写为$Y(p_{max})-Y(p_i)$，其反演公式仍可利用(7-5.14)式。

图 7-5-7 为典型的低速层分布的速度剖面，它可以分为三个反演单元，同时还给出其射线(实线)及回折点的轨迹(虚线)。

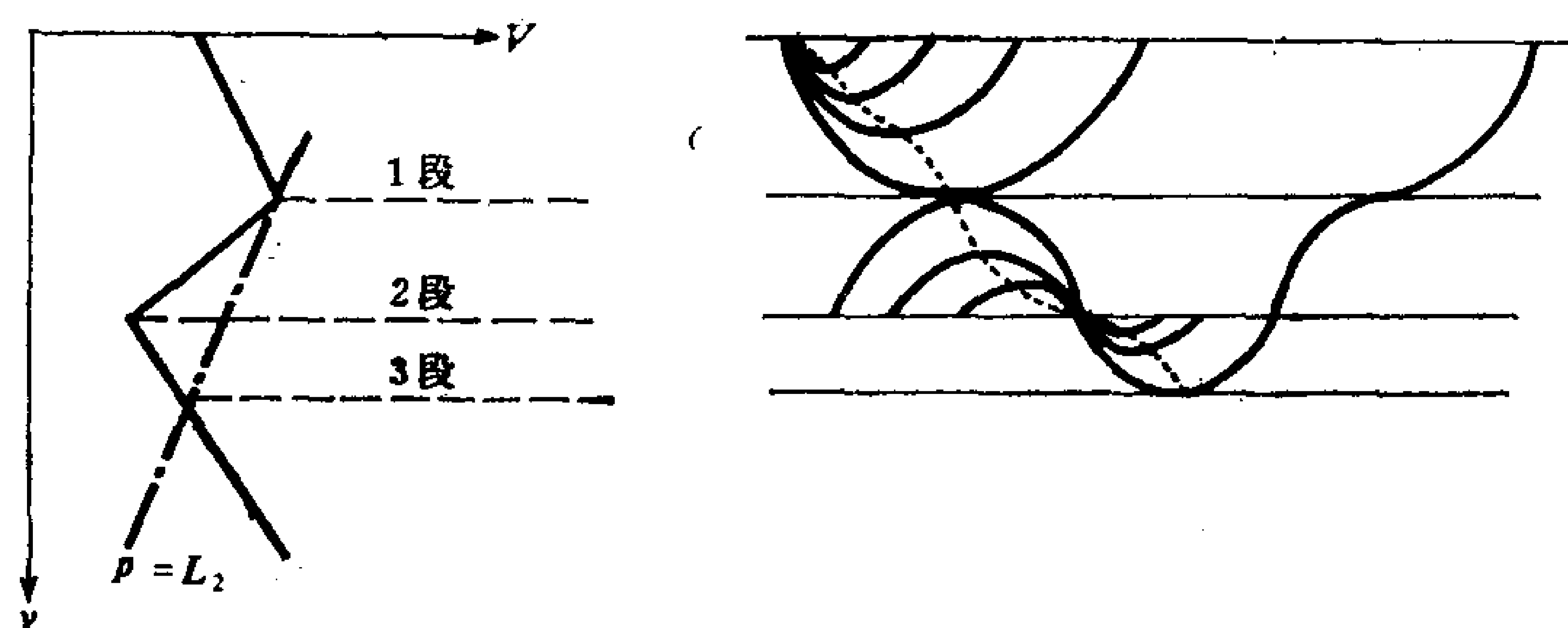



图 7-5-7

图 7-5-8(a) 为相应于7-5-7速度剖面的$p$值变化曲线。对每一个射线回折点，都有对应的X及$p$值。

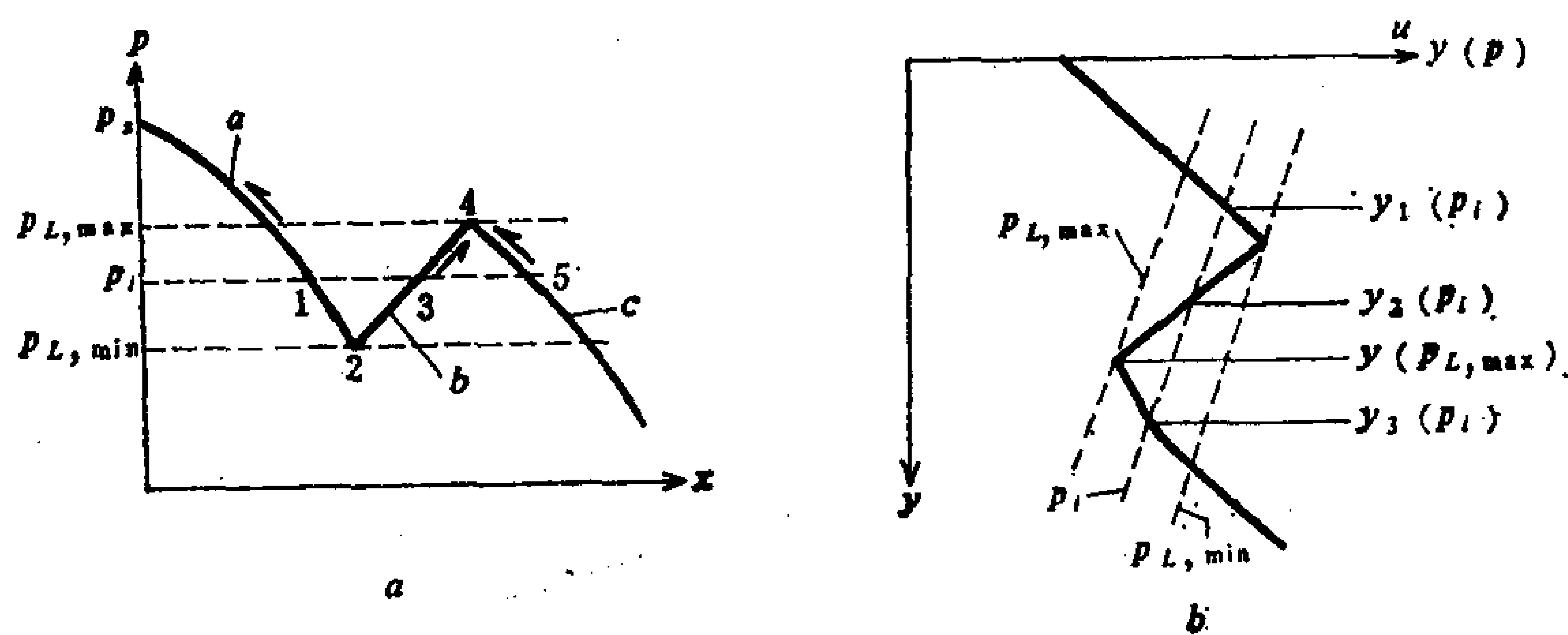



图 7-5-8

由已知的$p$—$X$曲线反演速度分布$u$—$Y$剖面可由上述反演单元获得。图 7-5-8 表示低速层反演计算过程。为了计算深度增量$Y_1(p_i)$—$Y(p_s)$，可以沿$p$-X曲线段$a$从 1 点到 2 点求积

$$Y_1(p_i)-Y(p_s)=\frac{2}{\pi}\int_{p_i}^{p_s}\frac{[X(p)-X(p_s)]}{(p^2-p_i^2)^{1/2}}dp. \qquad (7\text{-}5.15)$$

为了计算深度增量$Y(p_{L,max})-Y_2(p_i)$，可以沿曲线段$b$从 3 点到 4 点求积分

$$Y(p_{L,max})-Y_2(p_i)=\frac{2}{\pi}\int_{p_i}^{p_{Lmax}}\frac{[X(p_{L,max})-X(p)]}{(p^2-p_i^2)^{1/2}}dp.$$

(7-5. 16)

对于深度增量$Y_3(p_i)-Y(p_{L,max})$，可以沿 $p$-X 曲线段$c$从 5 点到 4 点求积分

$$Y_3(p_i)-Y(p_{L,max})=\frac{2}{\pi}\int_{p_i}^{p_{L,max}}\frac{[X(p)-X(p_{L,max})]}{(p^2-p_i^2)^{1/2}}dp.$$

(7-5.17)

由(7-5.16)及(7-5.17)式可得

$$Y_3(p_i)-Y_2(p_i)=\frac{2}{\pi}\int_{p_i}^{p_{L,max}}\frac{[X_c(p)-X_b(p)]}{(p^2-p_i^2)^{1/2}}dp.$$

(7-5.18)

式中$X_c(p)$表示$c$段曲线对应的$X$值，$X_b(p)$表示$b$段对应的$X$值。注意，$Y_3(p_i)$及$Y_2(p_i)$点在低速层内，而$Y_1(p_i)$点在低速层外。求出各点的$Y(p)$值后，不难由(7-5.9)式将各相应深度的速度$u(y)$求出。

## §7-6　理论走时曲线的计算

用观测走时曲线反演地下速度分布时，一般要选择地壳及上地幔的速度分层模型，计算理论走时曲线与实测曲线进行拟合。

这种拟合过程可借助于线性反演理论用迭代法实现，也可以人为选择多种模型进行比较。由于理论走时的计算对地球内部速度分布未加限制，故任意速度分布模型都能得到理论走时曲线。

设球对称层状地球模型如图7-6-1所示。$R_0$ 为地球半径，X 及$\theta$分别表示以公里或弧度（或度）为单位的震中距。设地球内部第$i$层的顶部$TOP$及底部$BOTTOM$ 的半径及速度分别为 $R_T$、$V_T$及$R_B$,$V_B$，该层的速度分布规律设为 $V=a_iR+b_i$。$R_M$及$V_{R_M}$分别表示射线穿透最深点的半径及该点的速度。

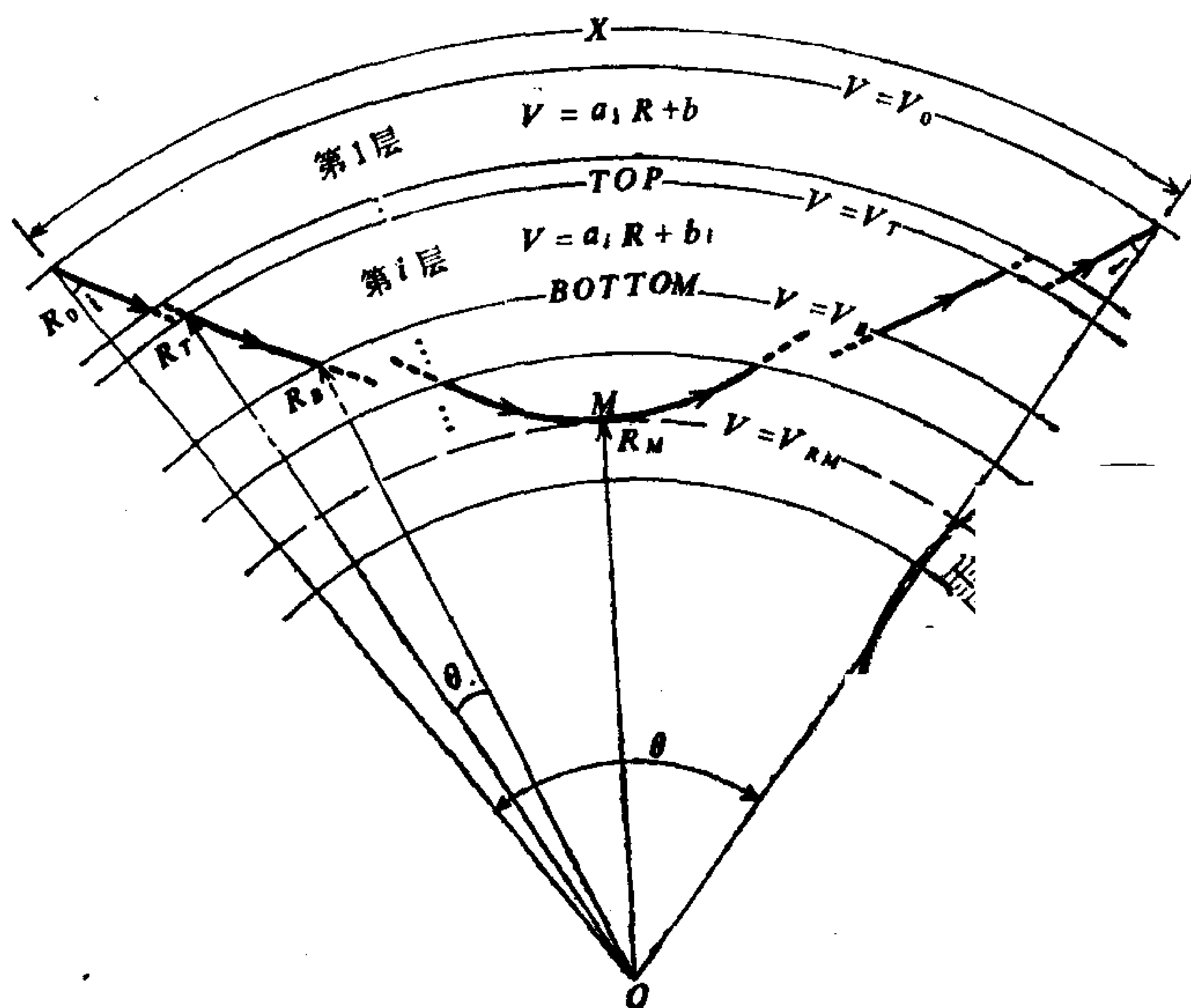



图 7-6-1

设某一条入射角为 $i$ 的射线通过每一层的单程震中距为 $\theta_i$，走时为$T_i$。并设该射线最深点$M$以上一共有 $n$ 层，于是该射线经过的距离及走时分别为

$$\theta=2\sum_{i=1}^{n}\theta_i,$$

$$T = 2\sum_{i=1}^{n} T_i. \tag{7-6.1}$$

式中$\theta$的单位是弧度(或度)，$T$的单位是秒。射线参数的定义与(7-1.5)式同，

$$p = \frac{R_0 \sin i}{V_0}.$$

$V_0$为地表速度，$R_0$是地球半径。

射线通过各层时所产生的单程震中距$\theta_i$及走时$T_i$的公式，与各层的速度变化规律有关。下面分别考虑三种情况.

## (一) 常 速 层

设某层速度为常数$V = b$，则$\theta_i$及$T_i$的计算公式为

$$\theta_i = \cos^{-1}(pb/R_T) - \cos^{-1}(pb/R_B),$$
$$T_i = \frac{1}{b}\left(\sqrt{R_T^2 - (pb)^2} - \sqrt{R_B^2 - (pb)^2}\right). \tag{7-6.2}$$

式中各符号的意义如前述。注意，射线能通过该层而回折的必要条件是$R_B \geqslant pb$。

## (二) 幂函数分布变速层

设某层的速度随半径$R$变化呈幂函数关系

$$V = aR^b. \tag{7-6.3}$$

式中$a$、$b$为常数。设该层顶部及底部的速度为$V_T$及$V_B$，半径为$R_T$及$R_B$，则可按下式算出$a$、$b$系数

$$b = \frac{\ln(V_T/V_B)}{\ln(R_T/R_B)},$$
$$a = \exp(\ln V_T - b\ln R_T). \tag{7-6.4}$$

令$\eta_T = R_T/V_T$，$\eta_B = R_B/V_B$，则穿透这一层的距离及走时分别为

$$\theta_i = \frac{1}{1-b}\left[\cos^{-1}(p/\eta_T) - \cos^{-1}(p/\eta_B)\right],$$

$$T_i = \frac{1}{1-b}\left[\sqrt{\eta_I^2 - p^2} - \sqrt{\eta_B^2 - p^2}\right]. \tag{7-6.5}$$

注意，射线能穿透该层而回折的必要条件是$\eta_B \geqslant p$。

## （三）线性函数分布变速层

设某层的速度随半径 $R$呈线性变化规律

$$V = aR + b. \tag{7-6.6}$$

由该层顶部及底部$V_T$、$V_B$及$R_T$、$R_B$可算出系数

$$a = \frac{(V_T - V_B)}{(R_T - R_B)},$$

$$b = V_T - aR_T. \tag{7-6.7}$$

令 $$c = 1 - a^2p^2,\ b^* = b/|b|. \tag{7-6.8}$$

当 $c$ 取 不 同 值时，计算公式是不同的。下面分三种情况讨论：

(1) $c>0$

$$\theta_i = \frac{ap}{\sqrt{c}}\Big[\ln\left(\sqrt{R_T^2 - P^2V_T^2} + R_T\sqrt{c} - \frac{abp^2}{\sqrt{c}}\right)$$

$$+ b^*\sin^{-1}\left(-apb^* - \frac{p|b|}{R_T}\right)$$

$$-\ln\left(\sqrt{R_B^2 - P^2V_B^2} + R_B\sqrt{c} - \frac{abp^2}{\sqrt{c}}\right)$$

$$- b^*\sin^{-1}\left(-apb^* - \frac{p|b|}{R_B}\right)\Big],$$

$$T = \frac{1}{|a|\sqrt{c}}\ln\left(|a|\sqrt{R_I^2 - p^2V_I^2} + V_T\sqrt{c} - \frac{b}{\sqrt{c}}\right)$$

$$+\frac{b^*}{|a|}\ln\left(\frac{|a|\sqrt{R_I^2 - P^2V_I^2} + |b|}{V_T} - b^*\right)$$

$$-\frac{1}{|a|\sqrt{c}}\ln\left(|a|\sqrt{R_B^2 - P^2V_B^2} + V_B\sqrt{c} - \frac{b}{\sqrt{c}}\right)$$

$$-\frac{b^*}{|a|}\ln\left(\frac{|a|\sqrt{R_B^2-P^2V_B^2}+|b|}{V_B}-b^*\right).$$

(7-6.9)

射线能穿透该层而回折的必要条件是$R_T \geqslant PV_T$及$R_B \geqslant pV_B$。

(2) $c=0$

$$\theta_i=-b^*\sqrt{-1-2\frac{aR_T}{b}}+b^*\sin^{-1}\left(-ab^*-\left|\frac{b}{a}\right|\frac{1}{R_T}\right)$$

$$+b^*\sqrt{-1-2\frac{aR_B}{b}}-b^*\sin^{-1}\left(-ab^*-\left|\frac{b}{a}\right|\frac{1}{R_B}\right),$$

$$T_i=\frac{-b^*}{|a|}\left[\sqrt{1-\frac{2V_T}{b}}+\ln\left(\left(\sqrt{1-\frac{2V_T}{b}}-1\right)\Big/\left(\sqrt{1-\frac{2V_T}{b}}+1\right)\right)\right.$$

$$\left.-\sqrt{1-\frac{2V_B}{b}}-\ln\left(\left(\sqrt{1-\frac{2V_B}{b}}-1\right)\Big/\left(\sqrt{1-\frac{2V_B}{b}}+1\right)\right)\right].$$

(7-6.10)

(3) $c<0$

$$\theta_i=\frac{ap}{\sqrt{-c}}\sin^{-1}\left(\frac{-cR_T}{p|b|}+apb^*\right)$$

$$+b^*\sin^{-1}\left(-apb^*-\frac{p|b|}{R_T}\right)$$

$$-\frac{ap}{\sqrt{-c}}\sin^{-1}\left(\frac{-cR_B}{p|b|}+apb^*\right)$$

$$-b^*\sin^{-1}\left(-apb^*-\frac{p|b|}{R_B}\right),$$

$$T_i=\frac{1}{|a|}\left[\frac{1}{\sqrt{-c}}\sin^{-1}\left(\frac{-cV_T+b}{p|ab|}\right)\right.$$

$$+b^*\ln\left(\frac{|a|\sqrt{R_T^2-p^2V_T^2}+|b|}{V_T}-b^*\right)$$

$$-\frac{1}{\sqrt{-c}}\sin^{-1}\left(\frac{-cV_B+b}{p|ab|}-b^*\ln\left(\frac{|a|\sqrt{R_B^2-p^2V_B^2}+|b|}{V_B}-b^*\right)\right].$$

(7-6.11)

下面讨论具体计算步骤。

步骤1：将地球划分为$n$个球层（$n+1$个界面），设每一界面的半径分别为$R_1$，$R_2\cdots$，$R_{n+1}$。对每一界面的波的传播速度分别赋以$V_1$，$V_2$，…，$V_{n+1}$。

步骤2：对每一层的速度变化规律赋以特征值。例如规定：1——常速层，2——幂函数变速层，3——线性函数变速层等。于是可以利用(7-6.4)及(7-6.7)等式分别计算常数$a$、$b$。

步骤3：从地表向地球深处按一定深度增量（例如$\Delta R=2$公里），分别计算该深度(半径$R$处)射线产生回折的射线参数

$$p=\frac{R}{V(R)}.$$

将该参数代入以上各层计算走时$T_i$及震中距$\theta_i$，再代入(7-6.1)式求和，即得穿透到半径$R$处的射线的走时$T$及距离$\theta$。当层层向下计算时，便得到理论走时曲线。

在计算过程中，如遇到低速层，则不符合产生回折射线的条件，出现影区。如遇高速梯度层，将会引起射线逆进回折。

## §7-7 一个数值反演实例

根据北京台网、大连、长春、牡丹江、海参崴等26个台站的记录，并选用从北京地区到阿留申群岛西端的300多个地震，经过椭圆扁率、震源深度及剥壳校正后(地壳平均厚度35公里)。得出了北京-萨哈林一线约48°长的$P$波走时曲线。

将校正后的走时曲线用Herglotz-Wiechert公式对上述观测曲线反演得到速度剖面，可见400公里650公里附近有明显的正

速度梯度带，另外在1200公里附近也有一较小的正速度梯度带。在80—100公里有低速层出现。如图7-7-1所示。

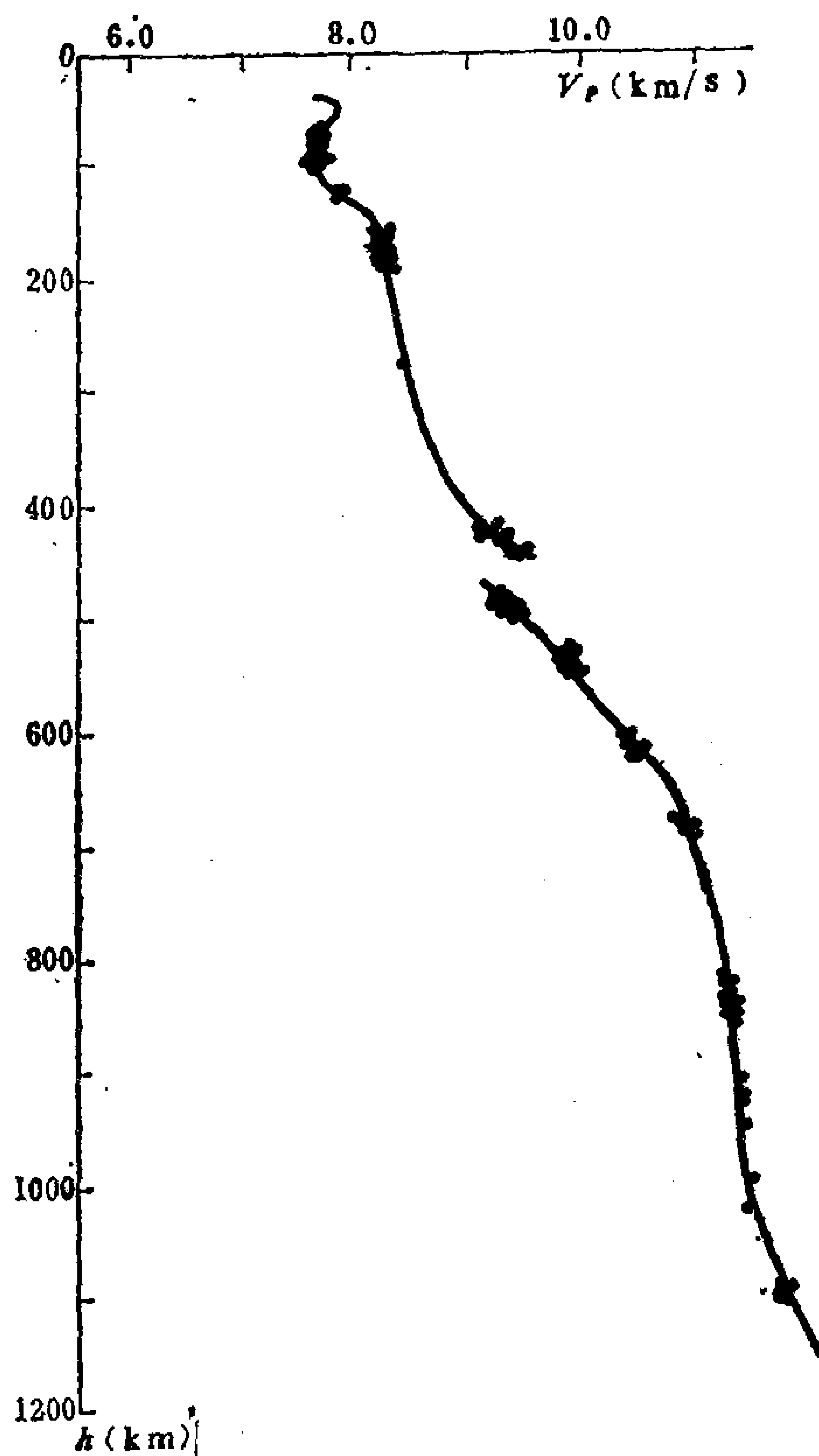



图 7-7-1

将上述$V$-$h$速度分布模型作为初值，反复调整此模型，计算理论走时曲线去拟合实测曲线。通过20多次修改，得出符合得最好的即为该区最后的速度模型。如图7-7-2示。

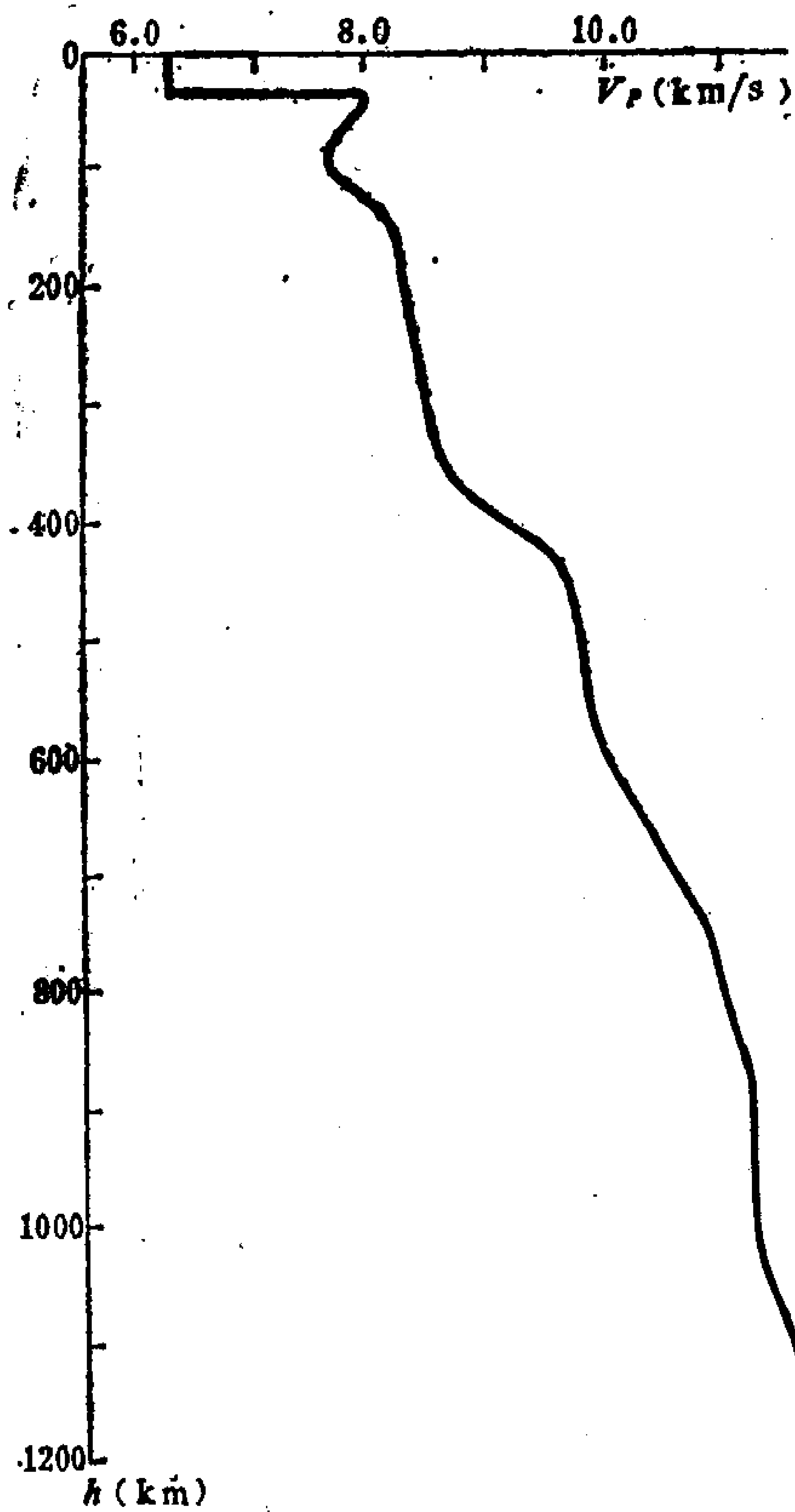



图 7-7-2

理论走时曲线与实际观测曲线的拟合情况如图 7-7-3 所示。

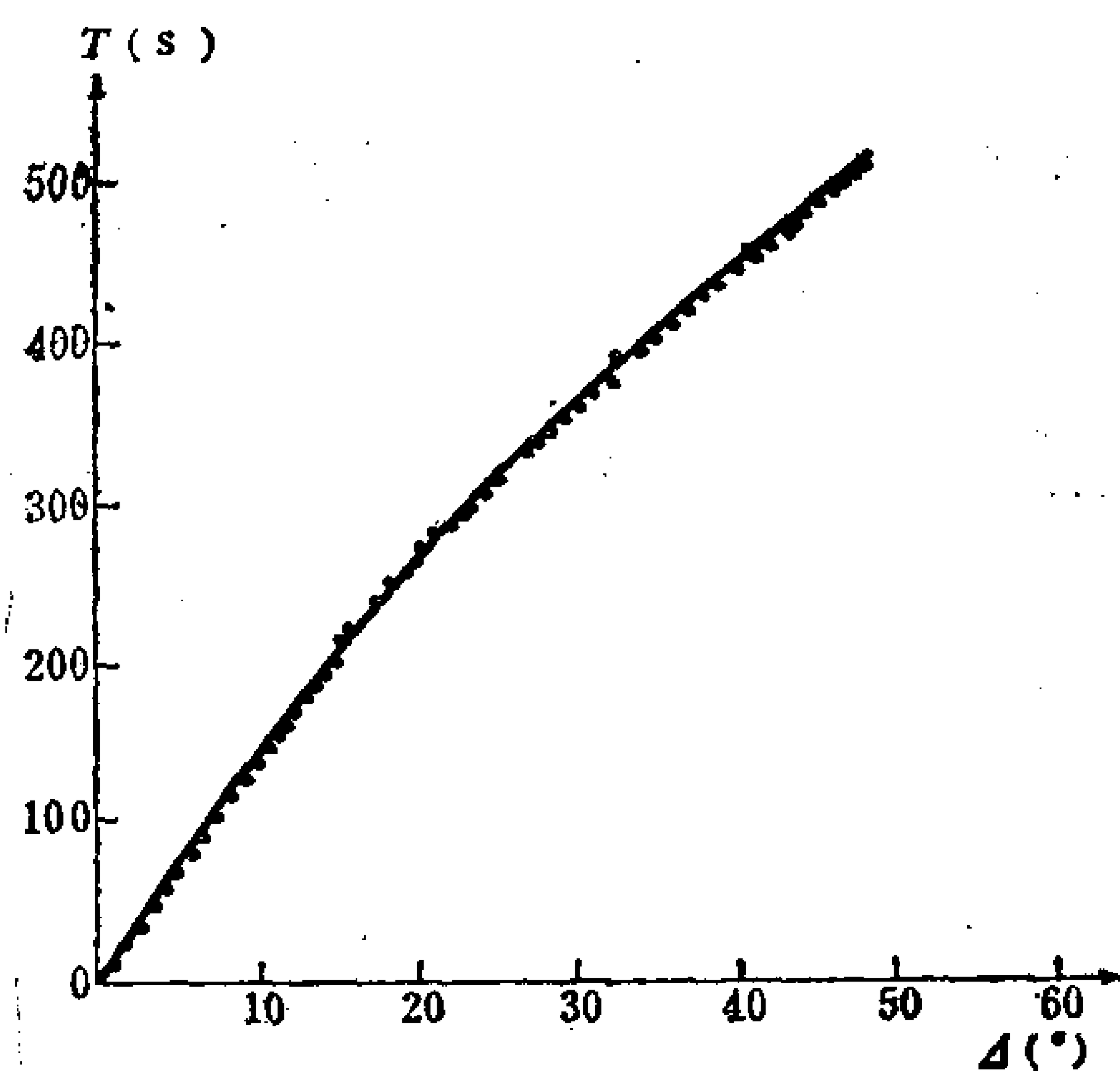



图 7-7-3

表7-7-1及表7-7-2分别为P波走时观测值及反演得到的最后速度分布模型，表7-7-3是体波走时反演的速度分布。

**表 7-7-1　　北京-萨哈林剖面P波走时观测值**

| Δ(°) | T(s) | Δ(°) | T(s) |
|---|---|---|---|
| 0.0 | 0.0 | 23.0 | 297.6 |
| 1.0 | 14.4 | 24.0 | 308.3 |
| 2.0 | 28.7 | 25.0 | 317.5 |
| 3.0 | 42.8 | 26.0 | 326.5 |
| 4.0 | 56.8 | 27.0 | 335.5 |
| 5.0 | 70.8 | 28.0 | 344.5 |
| 6.0 | 84.8 | 29.0 | 353.3 |
| 7.0 | 99.0 | 30.0 | 362.1 |
| 8.0 | 113.4 | 32.0 | 379.7 |
| 9.0 | 127.8 | 34.0 | 397.3 |
| 10.0 | 141.8 | 36.0 | 414.1 |
| 11.0 | 154.8 | 38.0 | 430.9 |
| 12.0 | 167.8 | 40.0 | 447.7 |

续表

| | | | |
|---|---|---|---|
| 13.0 | 180.8 | 42.0 | 464.2 |
| 14.0 | 193.8 | 44.0 | 479.5 |
| 15.0 | 206.8 | 46.0 | 494.5 |
| 16.0 | 219.8 | 48.0 | 509.5 |
| 17.0 | 232.8 | | |
| 18.0 | 243.6 | | |
| 19.0 | 254.4 | | |
| 20.0 | 265.2 | | |
| 21.0 | 276.0 | | |
| 22.0 | 286.8 | | |

表7-7-2　　北京-萨哈林剖面上地幔速度分布理论模型

| 度深$H$(km) | $V$(km/s) | 层速度特征值 |
|---|---|---|
| 35.0 | 7.8 | 2 |
| 55.0 | 7.9 | 2 |
| 80.0 | 7.6 | 2 |
| 120.0 | 8.1 | 2 |
| 160.0 | 8.2 | 2 |
| 180.0 | 8.3 | 1 |
| 200.0 | 8.35 | 2 |
| 300.0 | 8.5 | 2 |
| 400.0 | 9.2 | 3 |
| 440.0 | 9.6 | 2 |
| 480.0 | 9.7 | 2 |
| 530.0 | 9.8 | 2 |
| 600.0 | 10.0 | 3 |
| 740.0 | 10.9 | 2 |
| 910.0 | 11.25 | 2 |
| 1000.0 | 11.3 | 2 |

续表

| | | |
|---|---|---|
| 1060.0 | 11.45 | 2 |
| 1200.0 | 11.9 | 2 |
| 1300.0 | 12.25 | 2 |
| 1500.0 | 13.00 | |

特征值表示：1——常速层；2——幂函数分布；3——线性函数分布

**表7-7-3 北京-萨哈林测线体波走时反演的速度分布**

```
Programmed PAUSE input crh
    35.0
Programmed PAUSE input n, m
    40, 120
Programmed PAUSE input dll
    0.4
Programmed PAUSE input dlt
    0.5
```

| distance | time | distance | time |
|---|---|---|---|
| .0000 | .0000 | 1.0000 | 14.4000 |
| 2.0000 | 28.7000 | 3.0000 | 42.8000 |
| 4.0000 | 56.8000 | 5.0000 | 70.8000 |
| 6.0000 | 84.8000 | 7.0000 | 99.0000 |
| 8.0000 | 113.4000 | 9.0000 | 127.8000 |
| 10.0000 | 141.8000 | 11.0000 | 154.8000 |
| 12.0000 | 167.8000 | 13.0000 | 180.8000 |
| 14.0000 | 193.8000 | 15.0000 | 206.8000 |
| 16.0000 | 219.8000 | 17.0000 | 232.8000 |
| 18.0000 | 243.6000 | 19.0000 | 254.4000 |
| 20.0000 | 265.2000 | 21.0000 | 276.0000 |
| 22.0000 | 286.8000 | 23.0000 | 297.6000 |
| 24.0000 | 308.3000 | 25.0000 | 317.5000 |
| 26.0000 | 326.5000 | 27.0000 | 335.5000 |
| 28.0000 | 344.5000 | 29.0000 | 353.3000 |
| 30.0000 | 362.1000 | 32.0000 | 379.7000 |

| | | | |
|---|---|---|---|
| 34.0000 | 397.3000 | 36.0000 | 414.1000 |
| 38.0000 | 430.9000 | 40.0000 | 447.7000 |
| 42.0000 | 462.2000 | 44.0000 | 479.5000 |
| 46.0000 | 494.5000 | 48.0000 | 509.5000 |

| distance | time | depth | velocity |
|---|---|---|---|
| .2000 | .1470 | 35.0000 | 7.7853 |
| .6000 | 8.6690 | 35.1729 | 7.7864 |
| 1.0000 | 14.3489 | 35.6274 | 7.7883 |
| 1.4000 | 20.0267 | 36.3262 | 7.7907 |
| 1.8000 | 25.7018 | 37.2432 | 7.7937 |
| 2.2000 | 31.3736 | 38.3501 | 7.7971 |
| 2.6000 | 37.0420 | 39.4653 | 7.8001 |
| 3.0000 | 42.7075 | 40.5610 | 7.8026 |
| 3.4000 | 48.3704 | 41.5205 | 7.8044 |
| 3.8000 | 54.0316 | 42.1875 | 7.8052 |
| 4.2000 | 59.6922 | 42.4365 | 7.8054 |
| 4.6000 | 65.3527 | 42.4204 | 7.8050 |
| 5.0000 | 71.0138 | 42.5547 | 7.8036 |
| 5.4000 | 76.6759 | 42.7627 | 7.8022 |
| 5.8000 | 82.3384 | 42.8394 | 7.8020 |
| 6.2000 | 88.0006 | 43.1606 | 7.8028 |
| 6.6000 | 93.6608 | 46.4902 | 7.8034 |
| 7.0000 | 99.3161 | 51.4590 | 7.8060 |
| 7.4000 | 104.9634 | 57.5806 | 7.8124 |
| 7.8000 | 110.5982 | 64.9604 | 7.8238 |
| 8.2000 | 116.2158 | 73.3105 | 7.8407 |
| 8.6000 | 121.8115 | 82.5200 | 7.8633 |
| 9.0000 | 127.3805 | 92.5288 | 7.8920 |
| 9.4000 | 132.9183 | 102.9585 | 7.9255 |
| 9.8000 | 138.4224 | 113.5635 | 7.9625 |
| 10.2000 | 143.8904 | 124.3242 | 8.0025 |
| 10.6000 | 149.3219 | 134.6113 | 8.0422 |

| | | | |
|---|---|---|---|
| 11.0000 | 154.7184 | 144.4546 | 8.0808 |
| 11.4000 | 160.0812 | 153.8755 | 8.1179 |
| 11.8000 | 165.4123 | 162.8701 | 8.1531 |
| 12.2000 | 170.7136 | 171.5249 | 8.1864 |
| 12.6000 | 175.9862 | 180.1758 | 8.2193 |
| 13.0000 | 181.2305 | 188.8677 | 8.2519 |
| 13.4000 | 186.4464 | 167.9634 | 8.2857 |
| 13.8000 | 191.6322 | 207.7930 | 8.3224 |
| 14.2000 | 196.7857 | 218.4165 | 8.3625 |
| 14.6000 | 201.9035 | 230.3247 | 8.4085 |
| 15.0000 | 206.9810 | 243.5405 | 8.4611 |
| 15.4000 | 212.0134 | 258.1636 | 8.5213 |
| 15.8000 | 216.9953 | 274.2896 | 8.5904 |
| 16.2000 | 221.9210 | 291.8047 | 8.6686 |
| 16.6000 | 226.7858 | 310.2886 | 8.7545 |
| 17.0000 | 231.5860 | 329.7041 | 8.8481 |
| 17.4000 | 236.3191 | 349.1060 | 8.9447 |
| 17.8000 | 240.9874 | 367.6343 | 9.0386 |
| 18.2000 | 245.5942 | 385.3164 | 9.1290 |
| 18.6000 | 250.1440 | 401.8237 | 9.2135 |
| 19.0000 | 254.6419 | 417.1147 | 9.2911 |
| 19.4000 | 259.0930 | 431.3974 | 9.3625 |
| 19.8000 | 263.5012 | 444.8521 | 9.4285 |
| 20.2000 | 267.8702 | 457.5644 | 9.4894 |
| 20.6000 | 272.2025 | 470.1821 | 9.5487 |
| 21.0000 | 276.4990 | 482.7857 | 9.6070 |
| 21.4000 | 280.7601 | 495.8564 | 9.6668 |
| 21.8000 | 284.9844 | 509.8559 | 9.7307 |
| 22.2000 | 289.1696 | 524.8418 | 9.7995 |
| 22.6000 | 293.3130 | 541.2231 | 9.8755 |
| 23.0000 | 297.4109 | 559.0454 | 9.9596 |
| 23.4000 | 301.4599 | 578.1177 | 10.0514 |
| 23.8000 | 305.4571 | 598.2461 | 10.1503 |
| 24.2000 | 309.4001 | 619.2232 | 10.2554 |

| | | | |
|---|---|---|---|
| 24.6000 | 313.2888 | 639.6894 | 10.3594 |
| 25.0000 | 317.1257 | 659.4756 | 10.4605 |
| 25.4000 | 320.9137 | 678.1978 | 10.5563 |
| 25.8000 | 324.6571 | 695.4429 | 10.6437 |
| 26.2000 | 328.3606 | 711.1953 | 10.7223 |
| 26.6000 | 332.0288 | 725.4229 | 10.7917 |
| 27.0000 | 335.6665 | 738.0712 | 10.8514 |
| 27.4000 | 339.2781 | 749.2837 | 10.9024 |
| 27.8000 | 342.8675 | 759.1885 | 10.9453 |
| 28.2000 | 346.4382 | 767.7554 | 10.9805 |
| 28.6000 | 349.9935 | 775.2246 | 11.0092 |
| 29.0000 | 353.5360 | 781.5606 | 11.0319 |
| 29.4000 | 357.0684 | 787.0527 | 11.0499 |
| 29.8000 | 360.5924 | 792.0000 | 11.0649 |
| 30.2000 | 364.1090 | 796.4556 | 11.0772 |
| 30.6000 | 367.6190 | 801.0586 | 11.0890 |
| 31.0000 | 371.1223 | 805.8917 | 11.1009 |
| 31.4000 | 374.6187 | 810.9375 | 11.1128 |
| 31.8000 | 378.1082 | 816.1866 | 11.1247 |
| 32.2000 | 381.5906 | 821.7134 | 11.1368 |
| 32.6000 | 385.0652 | 827.9693 | 11.1504 |
| 33.0000 | 388.5312 | 835.0054 | 11.1658 |
| 33.4000 | 391.9875 | 842.8013 | 11.1829 |
| 33.8000 | 395.4332 | 851.3433 | 11.2020 |
| 34.2000 | 398.8672 | 860.5396 | 11.2227 |
| 34.6000 | 402.2890 | 869.9293 | 11.2441 |
| 35.0000 | 405.6982 | 879.4566 | 11.2657 |
| 35.4000 | 409.0650 | 889.1358 | 11.2876 |
| 35.8000 | 412.4792 | 898.9737 | 11.3099 |
| 36.2000 | 415.8505 | 909.0118 | 11.3325 |
| 36.6000 | 419.2087 | 919.4639 | 11.3560 |
| 37.0000 | 422.5532 | 930.3585 | 11.3806 |
| 37.4000 | 425.8835 | 941.6900 | 11.4063 |
| 37.8000 | 429.1991 | 953.4561 | 11.4331 |

| | | | |
|---|---|---|---|
| 38.2000 | 432.4995 | 965.6280 | 11.4611 |
| 38.6000 | 435.7842 | 978.0465 | 11.4897 |
| 39.0000 | 439.0531 | 990.6934 | 11.5189 |
| 39.4000 | 442.3058 | 1003.5730 | 11.5488 |
| 39.8000 | 445.5422 | 1016.6890 | 11.5793 |
| 40.2000 | 448.7622 | 1029.7350 | 11.6098 |
| 40.6000 | 451.9670 | 1040.8930 | 11.6352 |
| 41.0000 | 455.1595 | 1049.8820 | 11.6548 |
| 41.4000 | 458.3427 | 1056.6600 | 11.6685 |
| 41.8000 | 461.5196 | 1061.1170 | 11.6764 |
| 42.2000 | 464.6931 | 1063.6870 | 11.6799 |
| 42.6000 | 467.8635 | 1068.0830 | 11.6858 |
| 43.0000 | 471.0285 | 1074.7890 | 11.6952 |
| 43.4000 | 474.1858 | 1083.6920 | 11.7087 |
| 43.8000 | 477.3328 | 1094.7020 | 11.7263 |
| 44.2000 | 480.4676 | 1107.3400 | 11.7477 |
| 44.6000 | 483.5892 | 1119.2250 | 11.7682 |
| 45.0000 | 486.6990 | 1130.0780 | 11.7868 |
| 45.4000 | 489.7980 | 1139.9490 | 11.8033 |
| 45.8000 | 492.8874 | 1148.8530 | 11.8178 |
| 46.2000 | 495.9683 | 1156.7390 | 11.8299 |
| 46.6000 | 499.0422 | 1163.2850 | 11.8393 |
| 47.0000 | 502.1105 | 1168.4020 | 11.8459 |
| 47.4000 | 505.1750 | 1172.0170 | 11.8499 |
| 47.8000 | 508.2370 | 1173.9990 | 11.8514 |

# §7-8 计算程序

## (一) 球层介质体波走时正演程序

本程序用于计算任意分层的球对称地球中体波的走时曲线。使用者仅需给出地下各界面的深度及界面速度，以及各层的速度分布特征。即可得出完整的体波走时曲线。

（一）标识符及子程序的说明

NF——地球内部分界面数。

NF1——地球内部分层数（=NF-1）。

MC——从地面到最深处按半径间隔$\Delta R$所取的点数。

H0——剥壳深度（一般取平均地壳厚度）。

H(50)——存放地下界面深度的数组。

VP(50)——存放界面处波速的数组。

FACRAD(50)——界面到地球中心的距离。

FACVEL(50)——界面的波速。

RADI(2000)——回折点半径（按步长$\Delta R$由浅向深递增）。

VELI(2000)——相应于某射线回折点的速度。

DELI(2000)——相应于该射线的震中距$\Delta_i$。

TI(2000)——相应于该射线的到时$T_i$。

LK(50)——各层速度分布的特征数。各特征值的意义是：1——常速层；2——幂函数变速层；3——线性函数变速层。

RTB(50, 2)——各速度分层的上、下界面对应的半径$R_T$，$R_B$。

VTB(50, 2)各速度分层的上、下界面的速度值$V_T$、$V_B$。

A(50), B(50)——各速度分层相应的$a$，$b$系数。

EMERGE(XH, XVP)——计算球层模型$p$波走时曲线的子程序。虚元XH为各速度分层界面的深度数组，XVP为相应各界面的速度数组。

DELTRA(M, P, RTBM, VTBM, DSUM, TSUM)——计算震中距$\Delta_i$及走时$T_i$的子程序。虚元的意义如下：

M——射线回折点以上实际分层数。

P——回折射线的射线参数。

RTBM——各层上、下界面所相应的半径$R_T$，$R_B$。

VTBM——各层上、下界面所相应的速度$V_T$，$V_B$。

DSUM——对各层求和的震中距$\Delta$。

TSUM——对各层求和的走时$T$。

（二）FORTRAN-77语言程序

```
C       THIS FROGRAM IS DESIGNED FOR THE
C       FORWARD PROBLEM OF TRAVEL
C       TIME IN THE SPHERICAL LAYERS
C
        COMMON /N/NF, NF1, MC, HO
        COMMON /AB/A(50), B(50), LK(50),DELTA(1000),
           TRAVL(1000)
        DIMENSION H(50), VP(50)
        PAUSE'INPUT NF, MC, HO'
        READ *, NF, MC, HO
        NF1=NF-1
        PAUSE'INPUT H(I), VP(I), LK(I)'
        OPEN(1,FILE='DASPLY')
        READ(1, *)(H(I), VP(I), I=1, NF)
        READ(1, *)(LK(I), I=1, NF1)
        WRITE(*, 100)
  100   FORMAT)//30X, 'ORIGINAL DATA'//)
        WRITE(*, 150)(H(I), VP(I),I=1, NF)
  150   FORMAT(8F8.2)
        WRITE(*, 200)(LK(I), I=1, NF1)
  200   FORMAT(/4X, 20I4)
        CALL EMERGE(H, VP)
        END
C
        SUBROUTINE EMERGE(XH, XVP)
        COMMON /N/NF, NF1, MC, HO
        COMMON /AB/A(50), B(50),LK(50), DELTA(1000),
           TRAVL(1000)
        DIMENSION XH(50), XVP(50), FACRAD(50),
```

```
     FACVEL(50), AV(50), AR(50)
   DIMENSION RADI(2000), VELI(2000)
   DIMENSION RTB(50,2), VTB(50,2),RTBNEW(50,2)'
     VTBNEW(50, 2)
   DO 10 I=1, MC
10 RADI(I)=6371.0-HO-2.0*FLOAT(I-1)
   DO 20 I=1, NF
   FACRAD(I)=6371.0-XH(I)
   FACVEL(I)=XVP(I)
20 CONTINUE
   DO 30 I=1, NF1
   RTB(I, 1)=FACRAD(I)
   RTB(I, 2)-FACRAD(I+1)
   VTB(I, 1)=FACVEL(I)
   VTB(I, 2)=FACVEL(I+1)
30 CONTINUE
   DO 60 I=1, NF1
   IF(LK(I).EQ.1) GO TO 40
   IF(LK(I).EQ.2) GO TO 50
   A(I)=(VTB(I,1)-VTB(I,2))/(RTB(I,1)-RTB(I,2)
   B(I)=VTB(I, 1)-A(I)*RTB(I, 1)
   GO TO 60
40 A(I)=0.0
   B(I)=VTB(I, 1)
   GO TO 60
50 B(I)=ALOG(VTB(I, 1)/VTB(I, 2))/ALOG(RTB
     (I, 1)/RTB(I, 2))
   A(I)=EXP(ALOG(VTB(I, 1))-B(I)*ALOG
     (RTB(I, 1)))
60 CONTINUE
   DO 160 I=1, MC
   DO 100 J=1, NF1
```

```
      IF(RADI(I).LE.RTB(J, 1))M=J
100   CONTINUE
      IF(LK(M).EQ.1)  GO  TO140
      IF(LK(M).EQ.2)  GO  TO 150
      VELI(I)=A(M)*RADI(I)+B(M)
      GO  TO 160
140   VELI(I)=VTB(M, 1)
      GO  TO 160
150   VELI(I)=A(M)*RADI(I)**B(M)
160   CONTINUE
      WRITE(*, 210) (FACRAD(I), I=1, NF)
210   FORMAT(//30X, 'FACRAD'//(10F8.2))
      WRITE(*, 220) (FACVEL(I), I=1, NF)
220   FORMAT(//30X, 'FACVEL'//(10F8.2))
      WRITE(*, 230) ((RTB(I,J), VTB(I, J), J=1, 2),
         I=1, NF1)
230   FORMAT(//30X, 'THICKNESS AND VELOCITY',
         //(10F8.2))
      WRITE(*, 240)(A(I), B(I), I=1, NF1)
240   FORMAT(//,30X,'COEFFICIENT OF A AND B',
         //, (2E30.10))
      WRITE(*, 250) (RADI(I), VELI(I), I=1, MC)
250   FORMAT(//, 30X, 'RADI AND VELI', //, (6X,
         8F8.2))
      WRITE(*, 260)
260   FORMAT(/9X, 'NO', 5X, 'H', 8X, 'R', 11X, 'V',
         8X, 'P', 10X, 'DS', 8X, 'TS')
      DO 350 I=1, MC
      DO 310 J=1, NF1
      IF(RADI(I).LE.RTB(J, 1))M=J
310   CONTINUE
      DO 320 K=1, M
```

```
      DO 320 J=1, 2
      RTBNEW(K, J)=RTB(K, J)
      VTBNEW(K, J)=VTB(K, J)
320   CONTINUE
      RTBNEW(M, 2)=RADI(I)
      VTBNEW(M, 2)=VELI(I)
      P=RADI(I)/VELI(I)
      CALL DELTRA(M, P, RTBNEW, VTBNEW,DS,TS)
      IF(DS.EQ.0.0)  GO  TO 350
      HH=6371.0-RADI(I)
      WRITE(*, 330)I, HH,RADI(I),VELI(I),P,DS,TS
330   FORMAT(/, 6X, I4, 2X, F8.2, 2X, F8.2, 2X,F8.2,
         2X, F8.2, 2X, F8.2, 2X, F8.2)
350   CONTINUE
      RETURN
      END
C
      SUBROUTINE DELTRA  (M, P, RTBM, VTBM,
        DSUM, TSUM)
      COMMON  /AB/A(50), B(50), LK(50), DELTA(1000),
        TRAVL(1000)
      DIMENSION RTBM(50, 2), VTBM(50, 2)
      DSUM=0.0
        TSUM=0.0
      DO 200 I=1, M
      AI=A(I)
      BI=B(I)
      RT=RTBM(I, 1)
      RB=RTBM(I, 2)
      VT=VTBM(I, 1)
      VB=VTBM(I, 2)
      PVT=P*VT
```

```
PVB=P*VB
IF(RT.LT.PVT) GO TO 210
IF(RB.LT.PVB) GO TO 210
SPVT=SQRT(RT*RT-PVT*PVT)
SPVB=SQRT(RB*RB-PVB*PVB)
IF(LK(I).EQ.1) GO TO 100
IF(LK(I).EQ.2) GO TO 150
C=1.0-(AI*P)**2
BB=BI/ABS(BI)
ABA=ABS(AI)
ABB=ABS(BI)
ABAB=ABS(AI*BI)
IF(C.GT.0) GO TO 60
IF(C.LT.0) GO TO 80
BAI=-BB*AI
SBAI=-ABS(BI/AI)
BART=BAI+SBAI/RT
BARB=BAI+SBAI/RB
IF(ABS(BART).GT.1.0) GO TO 210
IF(ABS(BARB).GT.1.0) GO TO 210
ARBI=-1.0-2.0*AI*RB/BI
ARTI=-1.0-2.0*AI*RT/BI
IF(ARBI.LT.0.0) GO TO 210
IF(ARTI.LT.0.0) GO TO 210
 DSUM=DSUM+BB*(SQRT(ARBI)-SQRT(ARTI)
   +ASIN(BART)-ASIN(BARB))
VTBI=1.0-2.0*VT/BI
VBBI=1.0-2.0*VB/BI
IF(VTBI.LT.0.0) GO TO 210
IF(VBBI.LT.0.0) GO TO 210
DQVT=SQRT(VTBI)-1.0
SQVB=SQRT(VBBI)-1.0
```

```
      IF(SQVT.LE.0.0)  GO  TO 210
      IF(SQVB.LE.0.0)  GO  TO 210
      TSUM=TSUM-BB/ABA*(SQRT(VTBI)-SQRT
         (VBBI)
         1+ALOG(SQVT/(SQVT+2.0))-ALOG(SQVB/
         (SQVB+2.0)))
      GOTO 90
60    SC=SQRT(C)
      PAB=P*ABB
      APB=-AI*P*BB
      APRT=APB-PAB/RT
      APRB=APB-PAB/RB
      IF(ABS(APRT).GT.1.0)  GO  TO 210
      IF(ABS(APRB).GT.1.0)  GO  TO 210
      ABPS=AI*BI*P*P/SC
      SQTP=SPVT+RT*SC-ABPS
      SQBP=SPVB+RB*SC-ABPS
      IF(SQTP.LE.0.0)  GO  TO 210
      IF(SQBP.LE.0.0)  GO  TO 210
      DSUM=DSUM+AI*P/SC*(ALOG(SQTP)-ALOG
         (SQBP))+BB*(ASIN(APRT)-ASIN(APRB))
      BISC=BI/SC
      ASVT=ABA*SPVT+VT*SC-BISC
      ASVB=ABA*SPVB+VB*SC-BISC
      ASAT=(ABA*SPVT+ABB)/VT-BB
      ASAB=(ABA*SPVB+ABB)/VB-BB
      IF(ASVT.LE.0.0)  GO  TO 210
      IF(ASVB.LE.0.0.)  GO  TO 210
      IF(ASAT.LE.0.0)  GO  TO 210
      IF(ASAB.LE.0.0)  GO  TO 210
      TSUM=TSUM+1.0/(ABA*SC)*(ALOG(ASVT)
         -ALOG(ASVB))+BB/ABA*(ALOG(ASAT)-AL
```

```
      OG(ASAB))
    GO  TO 90
80  SC=SQRT(-C)
    PAB=P*ABB
    APB=AI*P*BB
    CRT=-C*RT/PAB+APB
    CRB=-C*RB/PAB+APB
    AIB=-AI*BB
    AIPT=(AIB-ABB/RT)*P
    AIPB=(AIB-ABB/RB)*P
    IF(ABS(CRT).GT.1.0)  GO  TO 210
    IF(ABS(CRB).GT.1.0)  GO  TO 210
    IF(ABS(AIPT).GT.1.0)  GO  TO 210
    IF(ABS(AIPB).GT.1.0)  GO  TO 210
    DSUM=DSUM+AI*P/SC*(ASIN(CRT)-ASIN
      (CRB))
      +BB*(ASIN(AIPT)-ASIN(AIPB))
    PBB=P*ABAB
    CVT=(-C*VT+BI)/PBB
    CVB=(-C*VB+BI)/PBB
    IF(ABS(CVT).GT.1.0)  GO  TO 210
    IF(ABS(CVB).GT.1.0)  GO  TO 210
    ASAT=(ABA*SPVT+ABB)/VT-BB
    ASAB=(ABA*SPVB+ABB)/VB-BB
    IF(ASAT.LE.0.0)GOTO 210
    IF(ASAB.LE.0.0)GOTO 210
    TSUM=TSUM+1.0/(ABA*SC)*(ASIN(CVT)-ASIN
      (CVB))+BB/ABA*(ALOG(ASAT)-ALOG
      (ASAB))
90  CONTINUE
    GO  TO 200
100 DSUM=DSUM-ASIN(P*BI/RT)+ASIN)P*BI /R
      B)
```

```
      TSUM=TSUM+(SQRT(RT*RT-(P*BI)**2)
          -SQRT(RB*RB(-P*BI)**2))/BI
      GOTO 200
150   ET=RT/VT
      EB=RB/VB
      DSUM=DSUM+(ASIN(P/EB)-ASIN(P/ET))/(1-BI)
      TSUM=TSUM+(SQRT(ET*ET-P*P)-SQRT(EB*
         EB-P*P))/(1-BI)
200   CONTINUE
      DSUM=2*DSUM*57.29578
      TSUM=2*TSUM
      GO  TO 200
210   DSUM=0.0
      TSUM=0.0
220   RETRNU
      END
```

## （二） 地球内部速度分布反演程序

本程序利用地面观测的体波走时，根据Herglotz-Wiechert方法反演地球内部纵向速度分布。为了取得较好的反演结果，对实测走时曲线按三次样条函数进行平滑处理。

**（一） 标识符及子程序说明**

N——体波走时曲线的实际测点数。

M——按点距(DLL)等分加密后的点数。

R0——地球平均半径，6371.0公里。

CRH——地壳厚度。

RC——剥壳后地球半径RC=6371-CRH。

C——工作单元。

D0——观测震中距，一维数组。

T0——走时观测值，一维数组。

TRA——用样条函数平滑插值得到的走时，一维数组。

DEL——按点距DLL加密得到的震中距，一维数组。

R——反演计算求得的射线穿透半径，一维数组。

V——反演计算求得的速度分布值，一维数组。

T——反演计算求得的相应射线的走时，一维数组。

D——反演计算求得的相应射线的震中距，一维数组。

DG——反演计算求得的相应射线的穿透深度，一维数组。

P——样条平滑插值求得的走时微商$dT/d\Delta$，一维数组。

PP——相应的走时二次微商$d^2T/d\Delta^2$，一维数组。

DLT——走时观测标准误差估计值。

SPLN(N, K, X, Y, DS, F, F1, F2)——三次样条函数平滑插值及求走时微商子程序。其中各形参为：

N——实测值结点个数。

K——插值点数，K＝2＊M—1。

X——存放结点横坐标(震中距)的一维数组。

Y——存放结点纵坐标(走时)的一维数组。

DS——存放插值点横坐标的一维数组。

F——存放插值点纵坐标的一维数组。

F1——存放插值点一次微商值的一维数组。

F2——存放插值点二次微商值的一维数组。

SM(N, X, Y, DY, S, N3, A, B, C, D)——样条函数平滑子程序，其中各形参为：

N, X, Y——同SPLN。

DY——第$i$个结点的相对数，可取$Y_i$的标准误差估计值，一维数组。

S——控制平滑程度的参数。

N3——工作单元。

A, B, C, D——存放三次样条函数各系数的一维数组。

**(二) FORTRAN-77 语言程序**

```
C     THIS PROGRAM IS DESIGNED FOR HERGLOTZ-
```

```
      WIECHERT INVERSE
C     IN THE MODEL OF SPHERICAL LAYERS
      DIMESION DO(100), TO(100), DEL(0:400), TRA
         (0:400), P(0:400),
         PP(0:400), R(0:200), T(0:200), V(0:200), D(0:
         200), DG(0:200)
      PAUSE 'INPUT CRH'
      R0=6371.0
      C=0.017452925
      READ*, CRH
      RC=R0-CRH
      PAUSE'INPUT N, M'
      READ*, N, M
      K=2*M-1
      OPEN(1, FILE='SEDAT')
      READ(1, *)(D0(I), I=1, N)
      READ(1, *)(T0(I), I=1, N)
      PAUSE 'INPUT DLL'
      READ*, DLL
      DO 10 I=0, K
10    DEL(I)=DLL*FLOAT(I)/2
      CALL SPLN(N, K, D0, T0, DEL, TRA, P, PP)
      WRITE(*, 80)
      WRITE(*, 100)(D0(I), TO(I), I=1, N)
      WRITE(*, 85)
      WRITE(*, 100)(DEL(I), TRA(I), P(I), PP(I),
         I=1, K)
      DO 30 I=1, M
      D(I)=DEL(2*I-1)
30    T(I)=TRA(2*I-1)
      DO 411 I=1, M
      PI=P(2*I-1)
```

```
      PI1=P(1)/IP
      Q=ABS(PI1*PI1-1.0)
      S=ALOG(PI1+SQRT(Q))
      IF(I.LT.2) GO  TO  9
      NN=2*(I-1)
      DO 50 J=2, NN, 2
      PJI2=P(J+1)/PI
      PJI1=P(J)/PI
      S1=PJI1*PJI1-1.0
      S2=PJI2*PJI2-1.0
      S1=PJI1+SQRT(ABS(S1))
      S2=PJI2+SQRT(ABS(S2))
   50 S=S+40*ALOG(ABS(S1))+2.0*ALOG(ABS(S2))
    9 S=S/6.0*DLL*C
      R(I)=RC/EXP(S/3.14159265)
      V(I)=(R(I)/PI)*C
  411 CONTINUE
      DO 60 I=1, M
   60 DG(I)=R0-R(I)
   80 FORMAT (//, 12X, 'DISTANCE', 12X, 'TIME',
         12X, 'DISTANCE', 12X,
      1'TIME', /)
   85 FORMAT(//, 12X, 'DISTANCE', 12X, 'TIME',
         16X, 'P', 16X, 'PP', /)
      WRITE (*, 90)
      WRITE(*, 100)(D(I), T(I), DG(I), V(I), I=1, M)
   90 FORMAT(//, 12X, 'DISTANCE', 12X,'TIME',14X,
         'DEPTH', 12X, VELOCITY', /)
  100 FORMAT(4F18.4)
      END
C
      SUBROUTINE SPLN(N, K, X, Y, DS, F, F1, F2)
```

```
      DIMENSION DY(100), A(100), B(100), C(100),
     *   D(100), X(100),
     *   Y(100), DS(0:400),F(0:400), F1(0:400), F2(0:400)
      S=N
      PAUSE 'INPUT DLT'
      READ* DLT
      DO 10 I=1, N
   10 DY(I)=DLT
      N1=N+1
      CALL SM(N, X, Y, DY, S, N1, A, B, C, D)
      F(1)=A(1)
      F1(1)=B(1)
      F2(1)=2.0*C(1)
      M5=2
      DO 20 I=2, K
   91 IF(DS(I).LT.X(M5)) GO  TO  93
      M5=M5+1
      IF(M5-N) 91, 91, 95
   93 M3=M5-1
      E=DS(I)-X(M3)
      F(I)=A(M3)+B(M3)*E+C(M3)*E*E+D(M3)*
     *  E*E*E
      F1(I)=B(M3)+2.0*C(M3)*E+3.0*D(M3)*E*E
   20 F2(I)=2.0*C(M3)+6.0*D(M3)*E
   95 CONTINUE
      RETURN
      END
C
      SUBROUTINE SM(N,X,Y,DY,S, N3, A, B, C, D)
      DIMENSION R(101), R1(101), R2(101), V(101),
     *   U(101), T(100), T1(100),
     *   X(100), Y(100), DY(100), A(100),B(100), C(100),
```

```
      D(100)
    R2(N)=0.0
    R2(N3)=0.0
    R1(N)=0.0
    R1(1)=0.0
    R(1)=0.0
    U(N)=0.0
    U(N3)=0.0
    U(1)=0.0
    P=0.0
    N1=N-1
91  H=X(2)-X(1)
    F=(Y(2)-Y(1))/H
    DO 10 I=2, N1
    G=H
    E=F
    H=X(I+1)-X(I)
    F=(Y(I+1)-Y(I))/H
    A(I)=F-E
    T(I)=2.0*(G+H)/3.0
    T1(I)=H/3.0
    R2(I)=DY(I-1)/G
    R(I)=DY(I+1)/H
    SGJ=1.0-DY(I)/G-DY(I)/H
10  R1(I)=SGJ-1.0
    DO 20 I=1, N1
    B(I)=R(I)**2+R1(I)**2+R2(I)**2
    C(I)=R(I)*R1(I+1)+R1(I)*R2(I+1)
20  D(I)=R(I)*R2(I+2)
    DO 30 I=2, N1
    IF(I.GT.2) GO TO 913
    R(I)=1.0/(B(I)+P*I(I))
```

```
      U(I)=A(I)
      GO TO 914
913   R1(I-1)=F*R(I-1)
      R2(I-2)=G*R(I-2)
      R(I)=1.0/(B(I)+P*T(I)-F*R1(I-1)-G*R2(I-2))
      U(I)=A(I)-R1(I-1)*U(I-1)-R2(I-2)*U(I-2)
914   F=C(I)+P*T1(I)-H*R1(I-1)
      G=H
30    H=D(I)
      DO 40 I=2, N1
      NI=N1-I+2
40    U(NI)=R(NI)*U(NI)-R1(NI)*U(NI+1)- R2(NI)
     * *U(NI+2)
      E=0.0
      H=0.0
      DO 50 I=1, N1
      G=H
      H=(U(I+1)-U(I))/(X(I+1)-X(I))
      V(I)=(H-G)*DY(I)
50    E=E+V(I)**2
      V(N)=U(N1)/(X(N)-X(N1))*DY(N)
      E=E+V(N)**2
      WW=S+0.000000001
      IF(E.LE.WW)  GO  TO 920
      F=0.0
      GG=0.0
      H=X(2)-X(1)
      G=H/3.0*(U(1)+2.0*U(2))
      DO 60 I=2, N1
      H1=H
      G1=G
      H=X(I+1)-X(I)
```

```
      G2=H/3.0*(2.0*U(I)+U(I+1))
      G=H/3.0*(U(I)+2.0*U(I+1))
      F=F+(G1+G2)*U(I)
      IF(I.EQ.2) GO TO 950
      IF(I.EQ.3) GO TO 951
      A(I)=(G1+G2-R1(I-1)*A(I-1)/SQRT(R(I-1))
         R2(I-2)*
         A(I-2)/SQRT(R(I-2)))*SQRT(R(I))
      GOTO 919
950   A(I)=(G1+G2)*SQRT(R(I))
      GOTO 919
951   A(I)=(G1+G2-R1(I-1)*A(I-1)/SQRT(R(I-1)))
         *SQRT(R(I))
919   GG=GG+A(I)**2
 60   CONTINUE
      P=P+(E-SQRT(S*E))/(F-P*GG)
      GOTO 91
920   DO 807 I=1, N
      A(I)=Y(I)-DY(I)*V(I)
      C(I)=P*U(I)
807   CONTINUE
      DO 90 I=1, N1
      H=X(I+1)-X(I)
      D(I)=(C(I+1)-C(I))/(3.0*H)
      B(I)=(A(I+1)-A(I))/H-(H*D(I)+C(I))*H
 90   CONTINUE
      RETURN
      END
```

## （三）有低速层分布时地球内部速度反演程序

本程序利用地面观测走时曲线计算射线参数 $p$，反演地球内部的纵向速度分布。本程序仅允许含一个低速层。

**（一）标识符及子程序**

N，M，CRH——与附录二同。

DLL——进行插值的加密点距。

D0，T0——与附录二同。

DET0——经插值加密后的点距(度)，一维数组。

DET——经低速层水平距离校正后的点距（度），一维数组。

P0——经插值加密后的走时曲线计算的射线参数(秒/度)，一维数组。

RL，VL——分别为低速层的半径及速度分布，一维数组。

DLT——走时观测标准误差估计值。

PL——低速层顶部的射线参数极大值。

DL——低速层顶部的深度。

K——低速层中的分层数。

HL——低速层中各分层的厚度，共$k$个层。

THL——低速层的总厚度，THL = ($k$ - 1) * HL。

VL——低速层中各薄层的速度分布，一维数组。

FL——震中距的积分函数。

DR——半径积分元$dr$。

DP——射线参数积分元$dp$。

NQ——计算射线最低点的个数。

RQ——射线最低点相应的半径。

HQ——射线最低点的深度(6371.0 - RQ)。

Q——射线参数$g$值。

LAGR——拉格朗日插值子程序。

SIMPSQ——辛普生积分子程序。

SPLN——三次样条函数平滑插值子程序。

SM——样条函数平滑子速序。

**（二）FORTRAN-77语言源程序：**

```
C
C       THIS PROGRAM IS DESIGNED FOR HERGLOTZ-
C          WIECHERT INVERSE
C       IN THE MODEL OF SPHERICAL LAYERS WITH
C       LVL
C
        COMMON/NM/N, M
        COMMON/DT/D0(300), T0(300), DET0(600),
           DET(600), P0(600)
        DIMENSION RL(30), VL(30), RL1(300), VL1(300),
           FL(300), D(1000), P(1000), F(1000)
        PAUSE 'INPUT N, M, CRH'
        READ*, N, M, CRH
        OPEN(1, FILE='SEDAT')
        READ(1, *)(D0(I), I=1, N)
        READ(1, *)(T0(I), I=1, N)
        PAUSE 'INPUT DLL'
        READ*, DLL
        WRITE(*, 5)N, M, CRH, DLL
      5 FORMAT(/,25X,'H-W INVERSION WITH LVL',/,
           15X, 'N=', I3, 6X, 'M='I3, 6X'CRH=', F4.1,
           6X, 'DLL=', F5.2/)
        DO 10 I=1, M
     10 DETO(I)=DLL*FLOAT(I-1)
        CALL SPLN
        WRITE(*, 15)
        WRITE(*, 100)(D0(I), T0(I), I=1, N)
     15 FORMAT(//, 12X, 'DISTANCE',12X,'TIME', 12X,
        'DISTANCE', 12X, 'TIME', /)
        WRITE(*, 80)
        WRITE(*, 100)(DETO(I), P0(I), I=1, M)
        PAUSE 'INPUT PL, DL'
```

```
      READ*, PL, DL
      B=180.0/3.141592653
      DO 20 I=1, M
      P0(I)=B*P0(I)
20    DET0(I)=DET0(I)/B
      PAUSE 'INPUT K, THL'
      READ*,K, THL
      PAUSE 'INPUT VL(I)'
      READ*, (VL(I), I=1, K)
      WRITE(*, 25)PL, DL, K, THL
25    FORMAT(/, 12X, 'PL=', F6.3, 6X, 'DL=', F6.2,
         6X, 'K=', I2, 6X, 'THL=', F6.2, /)
      WRITE(*, 28)(VL(I), I=1, K)
28    FORMAT(/, 25X, '--L-V-L--', /, 12X, 8F8.3)
      PL=PL*B
      HL=THL/FLOAT(K-1)
      RL(1)=6371.0-DL
      DO 30 J=2, K
30    RL(J)=RL(J-1)-HL
      DR=HL/10.0
      K1=(K-1)*10+1
      LR1(1)=RL(1)
      VL1(1)=VL(1)
      DO 40 J=2, K1
      RL1(J)=RL1(J-1)-DR
      RL2=RL1(J)
      CALL LAGR(K, RL, VL, RL2, VL2)
40    VL1(J)=VL2
      DO 333 I=1, M
      DET(I)=DET0(I)
      PP=P0(I)
      IF(PP.GE.PL)  GO  TO 333
```

```
      DO 50 J =1, K1
      X = RL1(J)/VL1(J)
      X = RL1(J) * SQRT(ABS(X * X - PP * PP))
 50   FL(J) = PP/X
      CALL SIMPSO(K1, DR, FL, S)
      S = 2.0 * S
      DET(I) = DET0(I) - S
333   CONTINUE
      N1 = IFIX((P0(1) - P0(M))/0.5)
      N1 = N1 + 1
      D(1) = DET(1)
      DO 70 I=1, N1
      P(I) = P0(1) - FLOAT(I-1) * 0.5
      P1 = P(I)
      CALL LAGR(M, P0, DET, P1, DETP)
 70   D(I) = DETP
      WRITE(*, 80)
      WRITE (*, 100)(D(I), P(I), I=1, N1)
      NQ = (N1 - 1)/10
      WRITE(*, 130)
      DO 555 I=1, NQ
      N2=10 * I+1
      Q = P(N2) - 0.01
      DO 90 J =1, N2
 90   F(J) = D(J)/SQRT((P(J) + Q) * )P(J) - Q))
      CALL SIMPSO (N2, 0.5, F, SO)
      SO = SO/3.141592653
      RQ = (6370.0 - CRH)/EXP(SO)
      IF(Q.GE.PL)  GO  TO 95
      RLB = RL(K)/RL(1)
      RQ = RQ * RLB
 95   HQ = 6371.0 - RQ
```

```
      VQ=RQ/Q
      WRITE(*, 120) I, Q, RQ, HQ, VQ
555   CONTINUE
 80   FORMAT (//,12X,'DISTANCE', 9X,'RAY P.' 11X,
     DISTANCE', 11X, 'RAY P.', /)
100   FORMAT(4F18.4)
120   FORMAT(15X, I4, 5X, F8.3, 5X, F8.2, 5X, F7.2,
         5X, F6.2)
130   FORMAT(//, 18X, 'NO', 8X, 'P0', 11X, 'R', 12X,
         'H', 11X, 'V')
      END
C
C
      SUBROUTINE LAGR (N, X0, Y0, X, Y)
      DIMENSION X0(600), Y0(600)
      I=1
 10   IF(X.GT.0.5*(X0(I+1)+X0(I+2)))  GO  TO 30
      IF(X.LE.0.5*(X0(N-2)+X0(N-1))) GO  TO 20
      I=I+1
      GO  TO 10
 20   I=N-2
 30   MA=I+2
      Y=0.0
      DO 6 J=I, MA
      PA=1.0
      DO 5 K=I, MA
      IF(J-K)4, 5, 4
  4   PA=PA*(X-X0(K))/(X0(J)-X0(K))
  5   CONTINUE
  6   Y=Y+PA*Y0(J)
      RETURN
      END
```

```
C
C
      SUBROUTINE SIMPSO (N, H, F, S)
      DIMENSION F(1000)
      S=F(1)+F(N)
      N1=N-1
      N2=N-2
      DO 2 I=2, N1, 2
    2 S=S+4.0*F(I)
      DO 3 I=2, N2, 2
    3 S=S+2.0*F(I)
      S=H*S/3.0
      RETURN
      END
C
      SUBROUTINE SPLN
      COMMON  /NM/N, M
      COMMON  /DT/DO(300), T0(300), DET0(600),
      DET(600), P0(600)
      COMMON /AB/DY(300),A(300),B(300), C(300),D(300)
      S=N
      PAUSE'INPUT DLT'
      READ*, DLT
      WRITE(*, 5) DLT
    5 FORMAT(/, 12X, 'DLT=', F5.2, /)
      DO 10 I=1, N
   10 DY(I)=DLT
      N1=N+1
      CALL SM(D0, T0, S, N1)
      DET(1)=A(1)
      P0(1)=B(1)
```

```
      M5=2
     DO 20 I=2, 2*M
  91 IF(DET0(I).LT.D0(M5)) GO TO 93
     M5=M5+1
     IF(M5-N)91, 91, 95
  93 M3=M5-1
     E=DET0(I)-D0(M3)
     DET(I)=A(M3)+B(M3)*E+C(M3)*E*E+D(M3)
    *E*E*E
  20 P0(I)=B(M3)+2.0*C(M3)*E+3.0*D(M3)*E*E
  95 CONTINUE
     RETURN
     END
C
C
     SUBROUTINE SM(X, Y, S, N3)
     COMMON/NM/N, M
     COMMON/AB/DY(300),A(300),B(300), C(300), D(300)
     DIMENSION R(300),R1(300),R2(300),V(300) U(300),
    T(300), T1(300), X(300), Y(300)
     R2(N)=0.0
     R2(N3)=0.0
     R1(N)=0.0
     R1(1)=0.0
     R(1)=0.0
     U(N)=0.0
     U(N3)=0.0
     U(1)=0.0
     P=0.0
     N1=N-1
  91 H=X(2)-X(1)
```

```
      F=(Y(2)-Y(1))/H
      DO 10 I=2, N1
      G=H
      E=F
      H=X(I+1)-X(I)
      F=(Y(I+1)-Y(I))/H
      A(I)=F-E
      T(I)=2.0*(G+H)/3.0
      T1(I)=H/3.0
      R2(I)=DY(I-1)/G
      R(I)=DY(I+1)/H
      SGJ=1.0-DY(I)/G-DY(I)/H
   10 R1(I)=SGJ-1.0
      DO 20 I=1, N1
      B(I)=R(I)**2+R1(I)**2+R2(I)**2
      C(I)=R(I)*R1(I+1)+R1(I)*R2(I+1)
   20 D(I)=R(I)*R2(I+2)
      DO 30 I=2, N1
      IF(I.GT.2) GO TO 913
      R(I)=1.0/(B(I)+P*T(I))
      U(I)=A(I)
      GO  TO 914
  913 R1(I-1)=F*R(I-1)
      R2(I-2)=G*R(I-2)
      R(I)=1.0/(B(I)+P*T(I)-F*R1(I-1)-G* R2(I-2))
      U(I)=A(I)-R1(I-1)*U(I-1)-R2(I-2)'*U(I-2)
  914 F=C(I)+P*T1(I)-H*R1(I-1)
      G=H
   30 H=D(I)
      DO 40 I=2, N1
      NI=N1-I+2
   40 U(NI)=R(NI)*U(NI)-R1(NI)*U(NI+1)-R2(NI)
```

```
     *U(NI+2)
    E=0.0
    H=0.0
    DO 50 I=1, N1
    G=H
    H=(U(I+1)-U(I))/(X(I+1)-X(I))
    V(I)=(H-G)*DY(I)
 50 E=E+V(I)**2
    V(N)=U(N1)/(X(N)-X(N1))*DY(N)
    E=E+V(N)**2
    WW=S+0.000000001
    IF(E.LE.WW) GO TO 920
    F=0.0
    GG=0.0
    H=X(2)-X(1)
    G=H/3.0*(U(1)+2.0*U(2))
    DO 60 I=2, N1
    H1=H
    G1=G
    H=X(I+1)-X(I)
    G2=H/3.0*(2.0*U(I)+U(I+1))
    G=H/3.0*(U(I)+2.0*U(I+1))
    F=F+(G1+G2)*U(I)
    IF(I.EQ.2) GO TO 950
    IF(I.EQ.3) GO TO 951
    A(I)=(G1+G2-R1(I-1)*A(I-1)/SQRT(R(I-1))
       -R2(I-2)*
     A(I-2)/SQRT(R(I-2)))*SQRT(R(I))
    GO TO 919
950 A(I)=(G1+G2)*SQRT(R(I))
    GO TO 919
951 A(I)=(G1+G2-R1(I-1)*A(I-1)/SQRT(R(I-1)))
```

```
       * SQRT(R(I))
919  GG=GG+A(I)* *2
 60  CONTINUE
     P=P+(E-SQRT(S*E))/(F-P*GG)
     GO  TO 91
920  DO 807 I=1, N
     A(I)=Y(I)-DY(I)*V(I)
     C(I)=P*U(I)
807  CONTINUE
     DO 90 I=1, N1
     H=X(I+1)-X(I)
     D(I)=(C(I+1)-C(I))/(3.0*H)
     B(I)=(A(I-1)-A(I))/(H-(H*D(I)+C(I))*H
 90  CONTINUE
     RETURN
     END
```

# 第八章　用地震面波研究地球内部结构

从本世纪五十年代以来，世界各国地震学家广泛地使用天然地震面波来研究地球内部的结构。这种方法利用天然地震的记录测定频散曲线，并将实测的频散曲线作反演计算，从而得到地球内部结构参数。

在反演计算时，必须将实测频散曲线与理论频散曲线相比较。尽管 Haskell 在五十年代初期就提出了计算任意多层地球模型理论频散曲线的矩阵方法。但在高速计算机问世之前，这种算法一直无法实现。当时仅能计算相当简单的地球模型(例如，单层或双层介质)。

过去，由地震记录获取面波频散曲线主要采用峰谷法，这是一种目视对比的手工操作法，精度低且易出错。1956年，Satô等人在面波频散计算中引入了Fourier 分析，使面波频散的测定建立在数值计算基础上。

自六十年代地震学各领域广泛使用数字计算机以来，面波研究工作有了突飞猛进的发展。首先，是把 Haskell 矩阵方法加以改进，在计算机上实现了多层面波频散曲线的快速计算(Schwab及knopoff等)。以后，利用数字滤波，快速 Fourier 变换等数学工具，也实现了用计算机测定面波频散曲线(Brune, Dziewonski 及 Landisman 等)。其它如普遍使用线性反演技术对面波频散曲线进行计算，得出了地壳上地幔的细结构(Dorman, Braile, Knopoff, Kovach等)。

随着用面波研究地球内部结构工作的广泛开展，发现了不同构造环境的地壳上地幔结构有很大差异。用面波研究地壳上地幔，不仅可以获得其分层的厚度及横波波速，还可以对地壳及上

地幔中的低速带作出较好的测定。

本章讨论面波频散的测定方法，以及用面波研究地球内部结构的正、反演方法。

## §8-1 地震面波及其频散

在远震记录图上，可以在 S波后观测到面波波列，这是沿着两种介质的分界面传播的波。当它沿介质的自由表面传播时，其能量主要集中在自由表面附近，并随着深度的增加能量迅速衰减。面波按其类型主要有Rayleigh波、Loue 波两类，下面简略讨论这两种波的形成及性质。

考虑一个均匀弹性半空间，当平面 SV 波大于临界角入射到自由表面时，反射p 波为沿着自由表面前进的不均匀波。这两种相互干涉，便产生了Rayleigh面波。

设$x$，$y$轴取在自由表面上，$z$ 轴垂直向下，在$z>0$的半空间有沿$x$方向传播的稳定振动。其位函数为

$$\phi = Ae^{-\bar{m}z}e^{i\omega\left(t-\frac{x}{c}\right)},$$

$$\psi = Be^{-\bar{k}z}e^{i\omega\left(t-\frac{x}{c}\right)}. \qquad (8\text{-}1.1)$$

式中c为简谐波同相面传播的速度，又称相速度。$\bar{m}>0$及$\bar{k}>0$，故$\phi$及$\psi$随深度$z$增加而衰减。

位函数$\phi$及$\psi$符合波动方程：

$$\nabla^2\phi = \frac{1}{\alpha^2}\frac{\partial^2\phi}{\partial t^2},$$

$$\nabla^2\psi = \frac{1}{\beta^2}\frac{\partial^2\psi}{\partial t^2}. \qquad (8\text{-}1.2)$$

式中$\alpha$及$\beta$分别为纵波波速及横波波速。

将(8-1.1)代入(8-1.2)式得

$$\bar{m}^2 = \frac{\omega^2}{\alpha^2}\left(\frac{\alpha^2}{c^2}-1\right),$$

$$\bar{k}^2 = \frac{\omega^2}{\beta^2}\left(\frac{\beta^2}{c^2} - 1\right). \qquad (8\text{-}1.3)$$

令 $k_1 = \alpha/\beta$ 及 $k_2 = c/\beta$，并根据边界条件 $\sigma_{zz} = 0$及$\sigma_{xz} = 0$，可以求出式(8-1.1)中待定系数 $A$、$B$ 具有非零解的条件为

$$(2 - k_2^2)^2 = 4\sqrt{1 - k_2^2}\sqrt{1 - \left(\frac{k_2}{k_1}\right)^2} \cdot \qquad (8\text{-}1.4)$$

因 $k_2 = c/\beta$，故上式实际上关于相速度$c$的方程，称为 Rayleigh 方程。当 $k_2$具有实数根时，表示Rayleigh波的存在。由(8-1.4)式可见，在(0,1)区间内$k_2$至少有一个实数根，且$k_2 < 1$。由此可知$c < \beta$，即自由表面的 Rayleigh 波相速度$c$总是小于横波速$\beta$的，其质点运动为逆进的椭圆。如图8-1-1示。

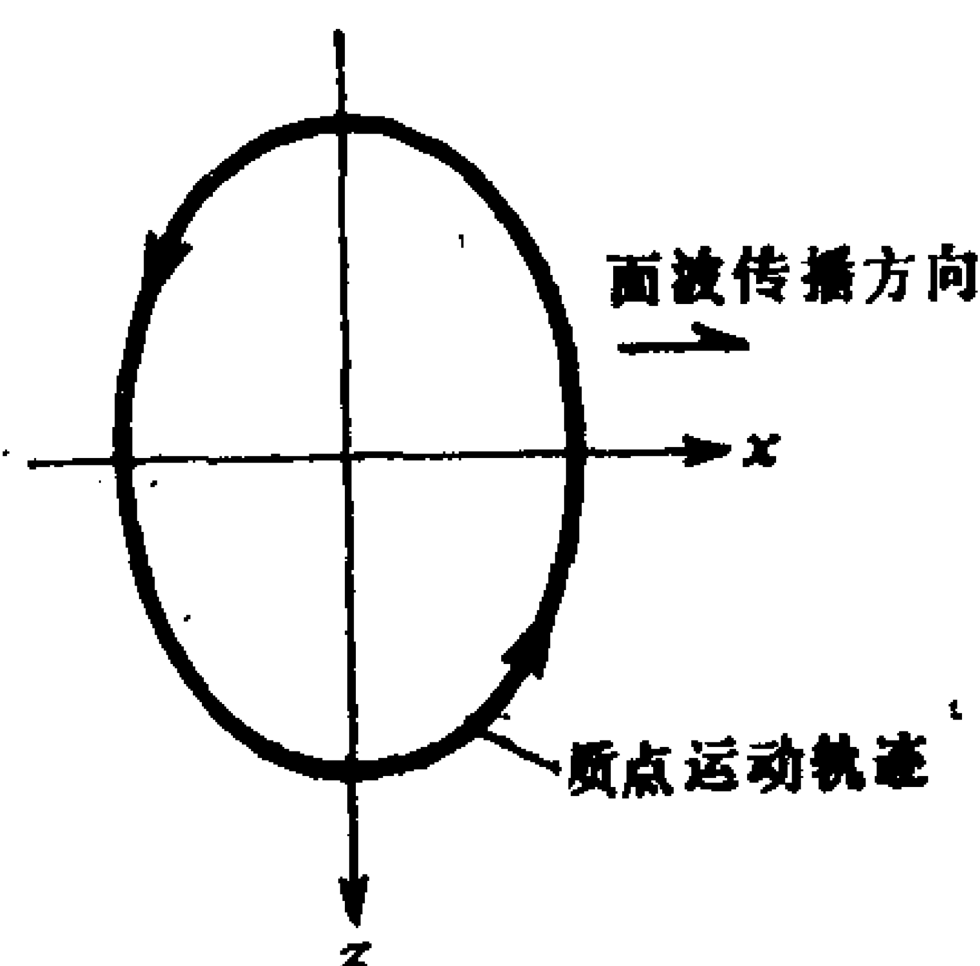



图 8-1-1

对于Poisson固体(弹性系数$\lambda = \mu$)而言，可以求得半无限空间中的相速度为$c = 0.9194\beta$，即 Rayleigh 面波传播速度略小于横波波速，且与频率$\omega$无关，不产生频散现象。

在二层及多层分层介质中，无论是Rayleigh面波和 Loue面波，其相速度都随频率变化。这种情况称为面波的频散。下面仅讨论二层介质的 Loue波。Loue 波是 $SH$ 型的横面波，其振动方向

与传播方向垂直。设大陆单层地壳模型复盖层各有关参数为$\mu_1$，$\beta_1$，$\rho_1$，厚度为$H$；下半无限空间有关参数为$\mu_2$，$\beta_2$，$\rho_2$，则Loue波相速度的频散方程为

$$\mathrm{tg}\left(Hk\sqrt{\frac{c^2}{\beta_1^2}-1}\right)=\left(\frac{\mu_2}{\mu_1}\right)\frac{\sqrt{1-\frac{c^2}{\beta_2^2}}}{\sqrt{\frac{c^2}{\beta_1^2}-1}}. \tag{8-1.5}$$

式中各参量的意义是：

$\mu_2/\mu_1$——盖层与半无限空间的剪切模量比。

$\beta_1$——盖层的横波速度。

$\beta_2$——半无限空间的横波速度。

$H$——盖层厚度。

$k=\frac{\omega}{c}=\frac{2\pi}{cT}=\frac{2\pi}{\lambda}$，称为波数，它表示单位距离内波的个数(波长$\lambda$的倒数)。它是一个有用的参量。

$c$——相速度，其范围为$\beta_1<c<\beta_2$。

方程式（8-1.5）给出了相速度 $c$ 与频率$\omega$的关系，这是一个非线性函数，可以用第六章讲述的方程求根法计算给定频率的相速度。

由于地震面波各个谐波波速的不同，所以在传播过程中，面波波形将发生变化。一般情况下是周期大的波先到，周期小的波后到，这种现象称为正频散(或频散)。反之，如果波列是周期先小后大，则称为负频散。描述频散波列的速度常用到相速度和群速度两个不同的概念。

在讨论 Rayleigh波和 Love 波时，对波的位函数(考虑一维情况)假定为下列简谐波形式，

$$\phi=Ae^{i(\omega t-kx)}. \tag{8-1.6}$$

其中，$\omega=k\frac{x}{t}=kc$，或$c=\frac{\omega}{k}$；$k$ 称为波数。$c$是简谐波同相

面的传播速度，称为相速度。

实际上，地震波是由许多简谐波迭加而成的，它们的频率各不相同，而且频率谱一般是连续函数。因此，当各种不同频率成分的波迭加时，如果波峰相遇就会使振幅增大。反之就会相互抵消而使振幅减小。迭加形成的大振幅传播的速度称为群速度。由于波的传播能量都集中于大振幅处，因此群速度也表示波的能量传播速度。下面来讨论群速度与相速度的关系。

根据 Fourier 积分定义，可以将地震波表示为连续频谱$g(\omega)$或 $g(k)$ 的下列积分形式：

$$\phi(x,\ t)=\int_{-\infty}^{\infty}g(\omega)e^{i(\omega t-kx)}d\omega, \tag{8-1.7}$$

或
$$\phi(x,\ t)=\int_{-\infty}^{\infty}g(k)e^{i(kx-\omega t)}dk. \tag{8-1.8}$$

式 $g(\omega)$ 及 $g(k)$ 分别表示以圆频率 $\omega$ 及波数 $k$ 为自变量的谱函数，而波的相位为

$$\theta=\omega t-kx. \tag{8-1.9}$$

每一简谐波都以一定的相速度 $c$ 传播，$c$ 是频率 $\omega$ 或波数 $k$ 的函数。不同频率成分的波在传播过程中相互干涉，有的增强，有的减弱。在相位角 $\theta$ 为常数的波数 $k_0$ 处，它们相互迭加而使振辐增强。$\theta$ 等于常数即满足条件

$$\left.\frac{d\theta}{dk}\right|_{k_0}=0.$$

或
$$\frac{d\theta}{dk}=\frac{d}{dk}(\omega t-kx)$$

$$=\frac{d\omega}{dk}t-x=0.$$

于是
$$\frac{x}{t}=\frac{d\omega}{dk}=\frac{d(kc)}{dk}$$

$$=\left(c+k\frac{dc}{dk}\right)\bigg|_{k_0}. \qquad (8\text{-}1.10)$$

上式表示波数为 $k_0$ 的波的大振幅经过时间 $t$ 后传到了 $x$ 处，因此$\frac{x}{t}$就是波数为 $k_0$ 的波的群速度 $U$，即

$$U=c+k\frac{dc}{dk}. \qquad (8\text{-}1.11)$$

由于波数

$$k=\frac{\omega}{c}=\frac{2\pi}{cT}=\frac{2\pi}{\lambda}。$$

所以

$$dk=-\frac{2\pi}{\lambda^2}d\lambda.$$

因而群速度又可以表示为

$$U=c-\lambda\frac{dc}{d\lambda}. \qquad (8\text{-}1.12)$$

通常用于计算群速度 $U$ 的式子可由(8-1.10)推得

$$U=\frac{d\omega}{dk}=\frac{d\omega}{d\left(\frac{\omega}{c}\right)}=\frac{d\omega}{\frac{d\omega}{c}-\frac{\omega}{c^2}dc}=\frac{c}{1-\frac{\omega}{c}\frac{dc}{d\omega}}. \qquad (8\text{-}1.13)$$

因为 $\omega/c=2\pi/Tc$，故上式又可以写为

$$U=\frac{c}{1+\frac{T}{c}\frac{dc}{dT}}. \qquad (8\text{-}1.14)$$

这是后面常用到的群速度计算公式。

关于面波频散的概念可由图8-1-2表示。图中给出一系列以 $t=0$为起点，按震中距排列的接收台站记录波形图。波列中每一个频率的波将以适当的群速度通过波列向前传播。对于给定的频

率此群速度将为一常数，频率不同群速度也不同。于是，对一系列接收台的波列给定频率的位置联线，将是通过原点的一条直线。此直线的斜率则是群速度。图中分别给出对应三个不同频率$\omega_1$，$\omega_2$及$\omega_3$相应的群速度$U_1$，$U_2$及$U_3$。$A$、$B$、$C$、$D$几条曲线分别表示同相位的联线。这些线上任一点的斜率即为该点某一特定频率的相速度值。

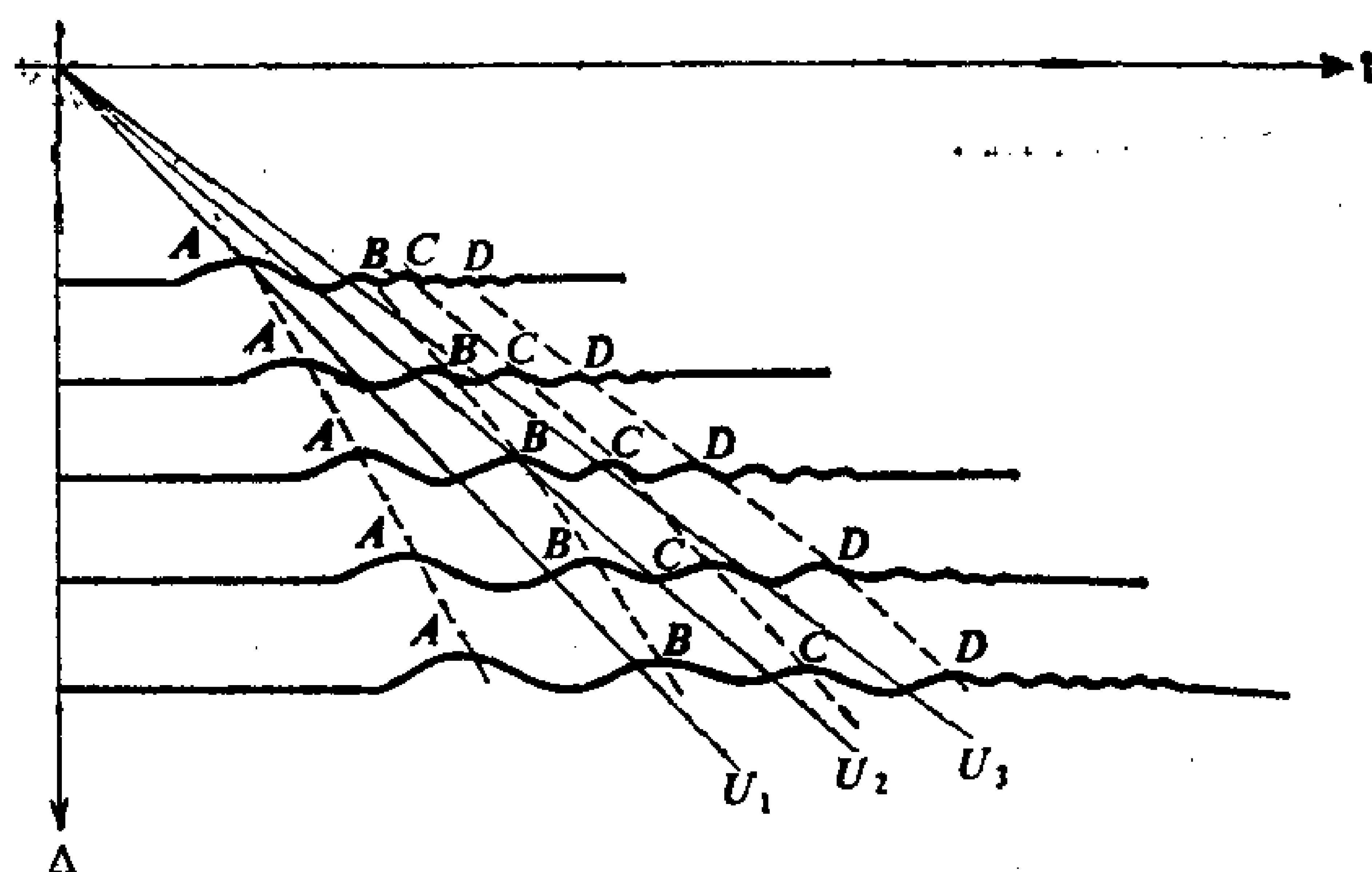



图 8-1-2

## §8-2 测定面波频散的计算方法

应用计算机测定面波频散曲线，不仅可以提高测定精度，还可以扩大测定的周期范围。因此，利用数字方法测定面波频散曲线已得到广泛的应用。

### （一）测定面波频散数据的预处理

面波波形的离散化，按$\Delta t = 1\text{s}$采样已能满足要求。为提高采样效率，实际上是用较大的不等距间隔采样，然后用插值法取得加密的等间隔样点。如果发现记录走纸速度不匀和记录基线不水平，则可按线性插值法作走纸速度和零点漂移校正。为了得

到真实的地动位移，还要作仪器频率特性校正。

由于 Rayleigh 波的质点位移与入射面一致，Loue 波的质点位移与入射面垂直。所以，在测定 Rayleigh 波频散时，一般只取垂直分量记录，而在测定 Loue 波频散时，则要将两水平分量作极化合成，以得到横向的振动分量。设地震面波入射方位角(以正北方向顺时针旋转为正)为 $\phi$，地震记录东西向及南北向的分量分别为 $A_E$、$A_N$，Loue 波的位移 $L$ 与 Rayleigh波水平分量位移 $R$ 可由下式计算：

$$\begin{pmatrix} L \\ R \end{pmatrix} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} A_E \\ A_N \end{pmatrix} . \tag{8-2.1}$$

## （二）群速度的测定

测量地震面波的群速度，主要是从地震记录中测量某一时刻的面波优势周期。由于地震台站到震中的距离 $\Delta$ 已知，发震时刻 $t_0$ 亦已知。而对应于某一优势周期（频率）的到时 $t$ 可以从数字化的地震波形上测定。因此群速度

$$U = \frac{\Delta}{t - t_0} . \tag{8-2.2}$$

对一系列优势周期 $T$ 进行测定群速度 $U$，则可以得到群速度频散曲线。

移动窗法及多重滤波法都是利用单台测定群速度的数值方法。前者在时间域内，后者在频率域内进行滤波。下面分述这两种算法。

### 1. 移动窗分析法

移动窗分析法就是沿着面波数字化信号 $x(t)$ 的记录时间，取一系列时间点 $t_i$，以 $t_i$ 时刻为中心乘以窗函数 $W(\tau)$，得到新的时间信号

$$Z_i(\tau)=x(t_i+\tau)W(\tau). \tag{8-2.3}$$

这一计算过程相当于时间窗沿面波信号移动，故称为移动窗分析法。

常用的时间窗为半余弦平方窗

$$W(\tau)=\begin{cases}\cos^2\dfrac{\pi\tau}{H_\omega} & |\tau|\leqslant\dfrac{H_\omega}{2},\\ 0 & |\tau|>\dfrac{H_\omega}{2}.\end{cases} \tag{8-2.4}$$

为了对所有的周期保持同样的分辨率，窗宽应随分析的周期增大而加宽。一般窗宽约为分析周期的五倍为宜。这是因为窗口太窄，周期分辨率较差。窗口太宽，则时间分辨率不好。

将(8-2.3)式进行Fourier变换，可以求得时窗中心对于时刻 $t_i$ 处不同周期 $T_k$ 的谱，

$$F(t_i,\ T_k)=\int_{-\frac{H_\omega}{2}}^{\frac{H_\omega}{2}} Z_i(\tau)e^{-j2\pi\tau/T_k}d\tau$$

$$=\int_{-\frac{H_\omega}{2}}^{\frac{H_\omega}{2}} Z_i(\tau)\cos(2\pi\tau/T_k)d\tau-j$$

$$\times\int_{-\frac{H_\omega}{2}}^{\frac{H_\omega}{2}} Z_i(\tau)\sin(2\pi\tau/T_k)d\tau$$

$$=\mathrm{Re}[F(t_i,\ T_k)]-j\,\mathrm{Im}[F(t_i,\ T_k)]. \tag{8-2.5}$$

因而求得其谱振幅为

$$A_{ik}=A(t_i,T_k)=\{\mathrm{Re}^2[F(t_i,T_k)]+\mathrm{Im}^2[F(t_i,\ T_k)]\}^{1/2};$$
$$i=1,\ 2,\ \cdots\cdots,\ N,\ k=1,\ 2,\ \cdots\cdots,\ M. \qquad (8\text{-}2.6)$$

$Ai_k$ 是由到时 $t_i$ 及周期$T_k$ 决定的谱振幅。由元素 $A_{ik}$ 构成的矩阵 $A_{NM}$，就把不同到时 $t$ 及不同周期 $T$ 全部的谱振幅包括在内。

以到时 $t_i$ 为纵坐标，周期$T_k$ 为横坐标，把矩阵 $A_{NM}$ 各元素打印在平面图上，将 $A_{ik}$ 元素最大值连线，即得所求的频散曲线。如图8-2-1示。

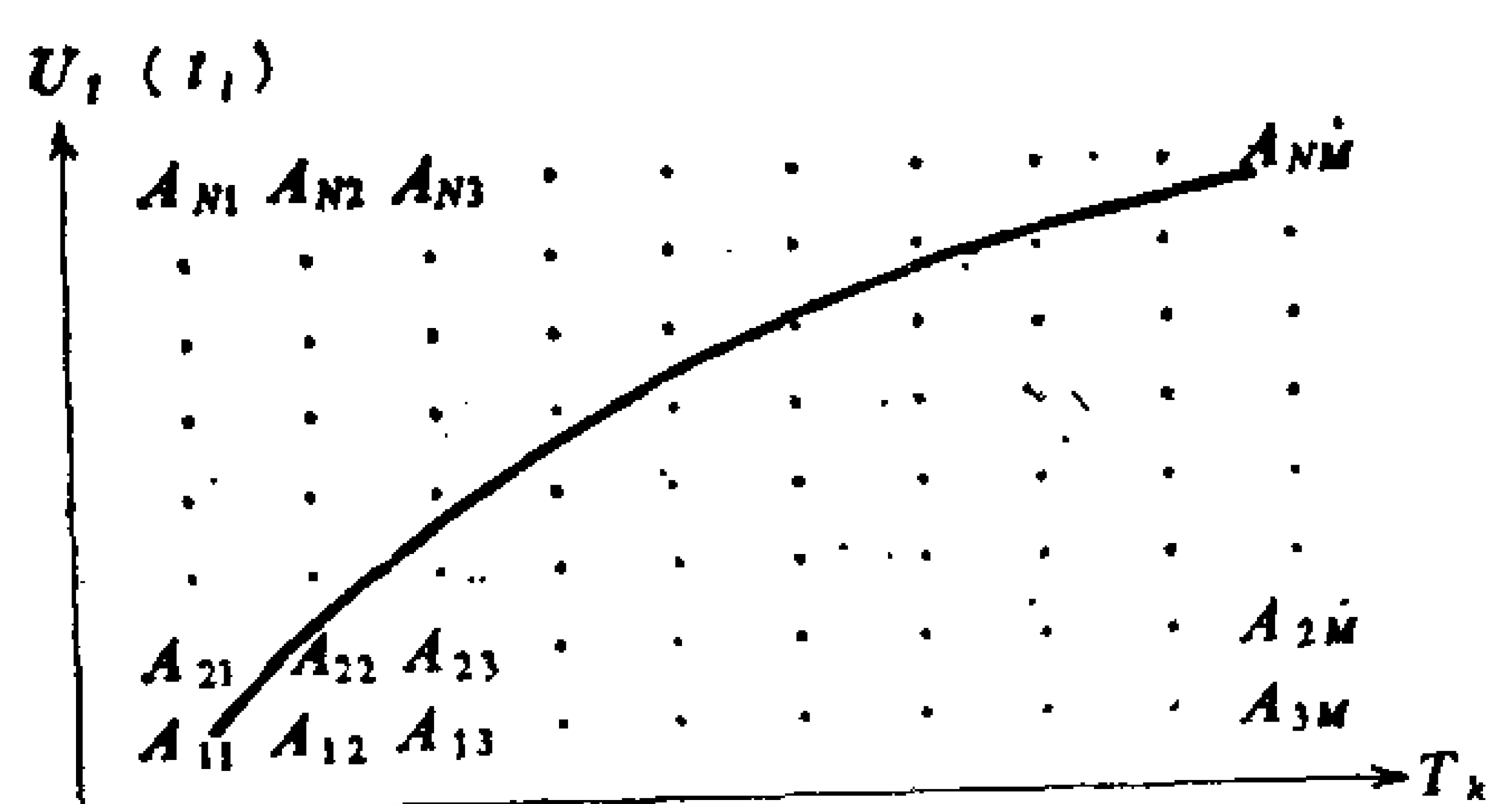



图 8-2-1

为了直接得到频散曲线，纵坐标实际上标群速度 $U_i$ 值。因为窗中距 $\Delta$ 和发震时刻 $t_0$ 是已知的，故由到时 $t_i$ 换算为群速度 $U_i$，可由以下简单公式表示

$$U_i=\frac{\Delta}{t_i-t_0},\qquad i=1,\ 2.\ \cdots.\ N. \qquad (8\text{-}2.7)$$

这样，平面坐标图上便直接给出群速度 $U$ 与周期 $T$ 的关系。

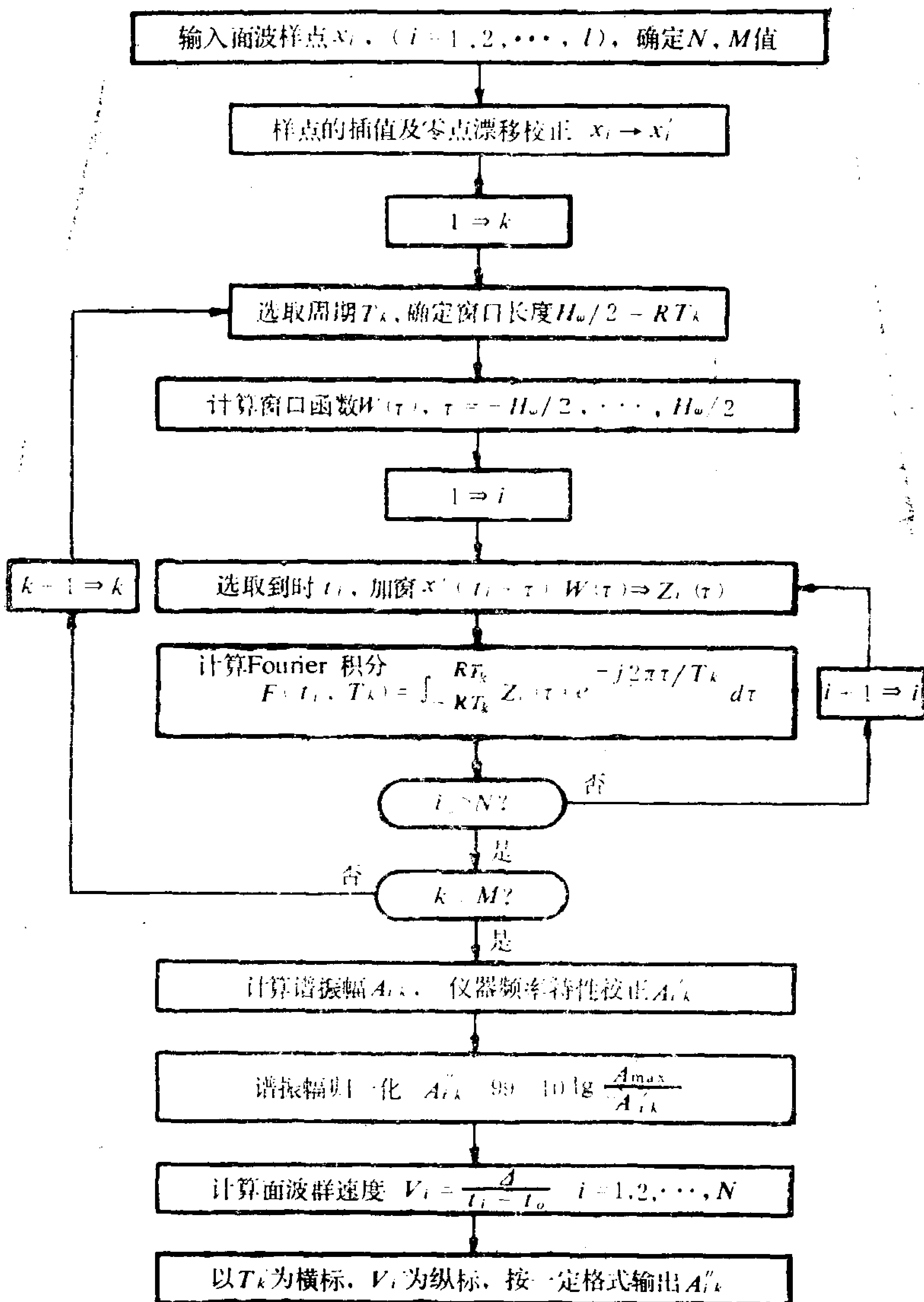
在计算谱振幅值时，对矩阵 $A_{NM}$ 按以下方式归一化：最大振幅 $A_{max}$ 取为99分贝，其它各相对振幅值为

$$A'_{ik}=99-10\lg\frac{A_{max}}{A_{ik}}. \qquad (8\text{-}2.8)$$

下面给出移动窗分析法的计算框图(图8-2-2)。

**2．多重滤波法**

如上所述，移动窗分析法是在时间域内将移动时间窗与面波信号 $x(t)$ 各点 $t_i$ 为中心，依次相乘，然后通过Fourier 变换求出该点不同频率的谱振幅。与此不同，多重滤波法是将时间信



输入面波样点 x_i，（i=1,2,···,I），确定N，M值
样点的插值及零点漂移校正 x_i → x_i'
1 ⇒ k
选取周期 T_k，确定窗口长度 H_ω/2 = RT_k
计算窗口函数 W(τ)，τ = −H_ω/2，···，H_ω/2
1 ⇒ i
选取到时 t_i，加窗 x'(t_i+τ)W(τ) ⇒ Z_i(τ)
计算Fourier 积分
F(t_i, T_k) = ∫_{−RT_k}^{RT_k} Z_i(τ) e^{−j2πτ/T_k} dτ
i+1 ⇒ i
k+1 ⇒ k
i > N?
否
是
k > M?
否
是
计算谱振幅 A_ik，仪器频率特性校正 A_ik'
谱振幅归一化 A_ik'' = 99 − 10 lg (A_max / A_ik')
计算面波群速度 V_i = Δ / (t_i − t_0)  i=1,2,···,N
以T_k为横标，V_i为纵标，按一定格式输出 A_ik''


图 8-2-2

号 $x(t)$ 经 Fourier变换为谱函数$X(f)$． 然后以不同的频率点为中心乘以谱窗函数，再经Fourier 逆变换，求出该频率在不同时刻的瞬时谱振幅。在各点频率上乘以谱窗函数，犹如进行不同频率的滤波，所以这种方法称为多重滤波。由于快速 Fourier 变换（FFT）算法的应用，使得多重滤波法的计算速度和效果都比移动窗分析法好，因此应用较为广泛。多重滤波法算法如下。

首先将面波信号 $x(t)$ 用 FFT 算法变换为频谱函数

$$
\begin{aligned}
X(f) &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \\
&= \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt - j\int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt \\
&= \mathrm{Re}[X(f)] - j\,\mathrm{Im}[X(f)]. \qquad (8\text{-}2.9)
\end{aligned}
$$

接着将谱函数 $X(f)$ 逐点乘以中心频率为 $f_k$ 的频率窗口 $Y(f_k, f)$.

$$S(f) = X(f)Y(f_k,\ f). \qquad (8\text{-}2.10)$$

频率窗口可采用 Gauss 函数

$$
y(f_k,\ f) = \begin{cases} 0 & f < (1-D)f_k, \\ e^{-\alpha\left(\frac{f-f_k}{f_k}\right)^2} & (1-D)f_k \leqslant f \leqslant (1+D)f_k, \\ 0 & f > (1+D)f_k. \end{cases} \qquad (8\text{-}2.11)
$$

式中，$f_k = 1/T_k$ 为频率窗口的中心频率；$T_k$ 是中心周期；$D$ 为频率窗口的相对带宽；$\alpha$ 为Gauss 函数峰值的锐度参数。实际计算表明，一般 $D$ 取为0.25，$\alpha$ 取为30—50效果较好。

对(8-2.10)式作 Fourier 逆变换，

$$h(t_i,\ f_k) = \int_{(1-D)f_k}^{(1+D)f_k} X(f)Y(f_k,\ f)e^{j2\pi ft_i}df$$

$$=\int_{(1-D)f_k}^{(1+D)f_k} X(f)Y(f_k,f)\cos(2\pi f t_i)df$$

$$+j\int_{(1-D)f_k}^{(1+D)f_k} X(f)Y(f_k,f)\sin(2\pi f t_i)df$$

$$=\mathrm{Re}[h(t_i,f_k)]+j\mathrm{Im}[h(t_i,f_k)];$$

$$i=1,2,\cdots,N,\ k=1,2,\cdots,M. \qquad (8\text{-}2.12)$$

最后，计算时间序列$h(t_i, f_k)$对某一中心频率为$f_k$在时间为$t_i$的振幅：

$$A_{i,k}=A(t_i,f_k)=(\mathrm{Re}^2[h(t_i,f_k)]+\mathrm{Im}^2[h(t_i,f_k)])^{1/2};$$

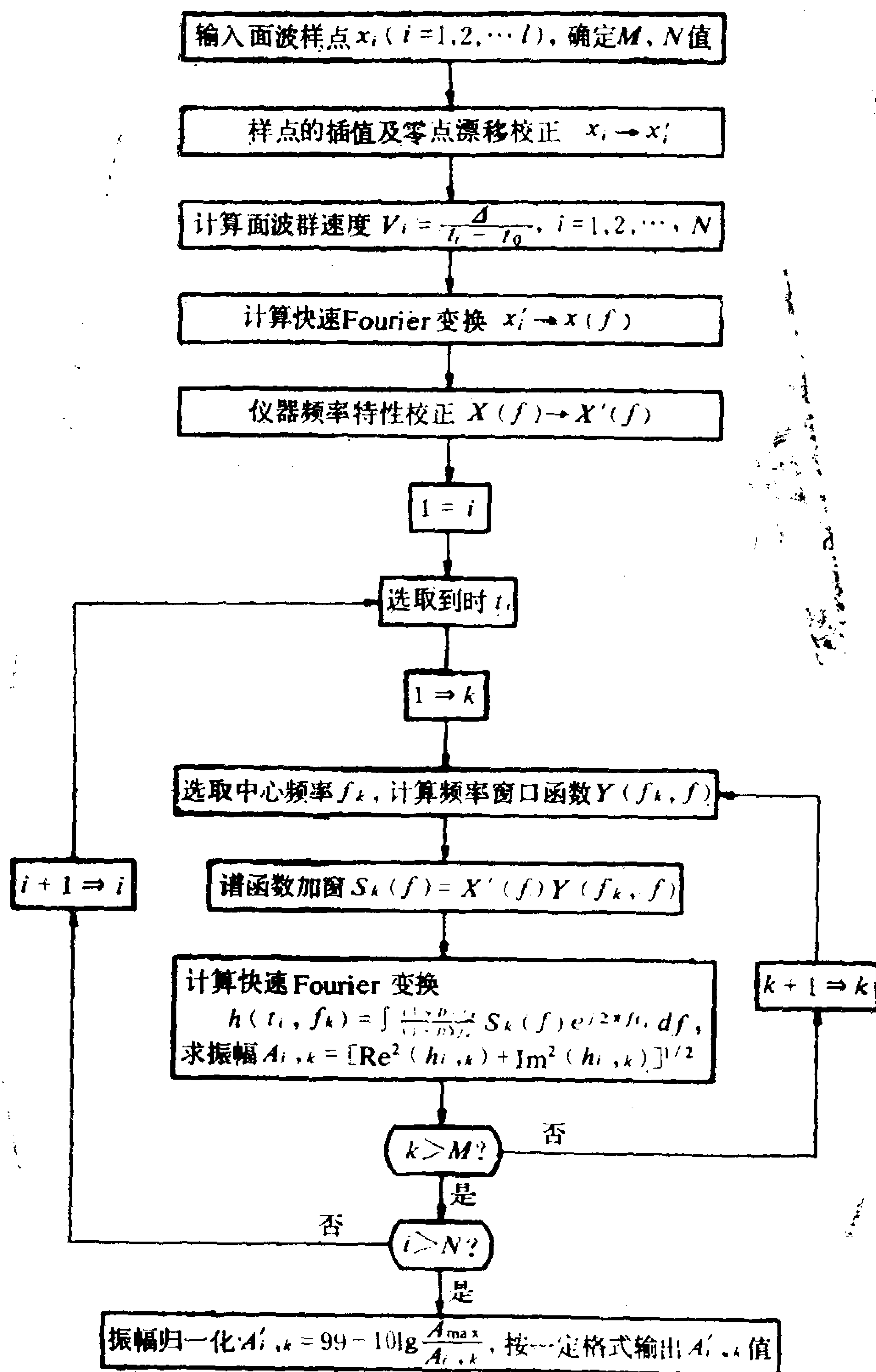
$$i=1,2,\cdots,N,\ K=1,2,\cdots,M. \qquad (8\text{-}2.13)$$

由此，可得到关于到时$t_i$和频率$f_k$的二维数组。与移动窗分析法同样，最后以群速度$U_i$为纵坐标，周期$T_k(=1/f_k)$为横坐标，将经过归一化后的$A_{i,k}$值输出，其最大值连线，即是群速度的频散曲线。图8-2-3为多重滤波法计算框图。

## （三）相速度的测定

求面波相速度早期的方法是Press于1956年提出的三台法。从理论上来说，只要震源产生的相位及仪器的相位移已知，用单台记录便可求得震中与记录台之间介质的相速度。这种方法首先由Sato引出。

若地震只激发基阶面波，在震源距为$r$，方位角为$\theta$的台站所记录到的信号为$x(t)$；注意，$t<0$时，$x(t)=0$，则Fourier变换为



输入面波样点$x_i$($i$=1,2,…$l$),确定$M$、$N$值
样点的插值及零点漂移校正 $x_i \to x_i'$
计算面波群速度 $V_i = \frac{\Delta}{t_i - t_0}$, $i$=1,2,…,$N$
计算快速Fourier变换 $x_i' \to x(f)$
仪器频率特性校正 $X(f) \to X'(f)$
$1 = i$
选取到时$t_i$
$1 \Rightarrow k$
选取中心频率$f_k$,计算频率窗口函数$Y(f_k, f)$
谱函数加窗 $S_k(f) = X'(f) Y(f_k, f)$
计算快速Fourier 变换
求振幅$A_{i,k} = [\mathrm{Re}^2(h_{i,k}) + \mathrm{Im}^2(h_{i,k})]^{1/2}$
$k+1 \Rightarrow k$
$i+1 \Rightarrow i$
$k > M$?
否
是
$i > N$?
否
是
振幅归一化$A'_{i,k} = 99 - 10\lg\frac{A_{max}}{A_{i,k}}$,按一定格式输出$A'_{i,k}$值


图 8-2-3

$$X(f)=\int_0^{\infty}x(t)e^{-j2\pi ft}dt. \tag{8-2-14}$$

信号的复数谱 $X(f)$ 可以由振辐谱和相位谱来表示，

$$X(f)=A(r,\ \theta,\ f)e^{j\phi(r,\theta,f)}. \tag{8-2-15}$$

上式中相位函数可以写为

$$\phi(r,\ \theta,\ f)=k(f)r+\phi_0(\theta,\ f)+\phi_i(f)2n\pi. \tag{8-2-16}$$

其中，$k(f)$ 为波数（即 $k=2\pi/\lambda$），$\phi_0(\theta,\ f)$ 为震源处的初相，$\phi_i(f)$ 为记录仪器的相移。因为 Fourier 变换所得的相位角总是在0与 $2\pi$ 之间，它与实际的相位角可能差 $2n\pi$ $(n=0,1,2,\cdots)$。为了能够估计 $k(f)$，必须确定 $\phi_0(\theta,\ f)$，$\phi_i(f)$及整数 $n$。一般说来，仪器相移 $\phi_i$ 可根据仪器特性计算出来。但很多情况下，作面波分析用的地震震源机制难于测定，所以 $\phi_0$ 也不确定。单台法的应用受到限制。

如果位于同一方位角上有两个台站的记录，则可以避免因震源机制不确定引起的困难。下面讨论两种计算方法，以测定两台间面波相速度的方法。

1．相位差法

设有位于同一方位角上的两台记录，由(8-2.16)式不难得

$$k(f)r_1=\phi_1(r,\ \theta,\ f)-\phi_0(\theta,\ f)-\phi_{i1}(f)-2n\pi;$$
$$k(f)r_2=\phi_2(r,\ \theta,\ f)-\phi_0(\theta,\ f)-\phi_{i2}(f)-2m\pi. \tag{8-2-17}$$

将二式相减，并经过简单的变换，可得到两台间介质的相速度

$$c(f)=\frac{2\pi f}{k(f)}=\frac{2\pi f(r_2-r_1)}{\phi_2-\phi_1-\phi_{i2}+\phi_{i1}+2l\pi}. \tag{8-2-18}$$

若两台仪器频率特性相同（$\phi_{i2}=\phi_{i1}$），并将频率 $f$ 改为周期 $T$

表示，则上式可简化为

$$c(T)=\frac{r_2-r_1}{\frac{T}{2\pi}\ (\phi_2-\phi_1+2l\pi)} \tag{8-2-19}$$

上式中，整数 $l$ 可以根据相速度的粗略范围而估计。所以，由此测定的相速度是不唯一的。不同的估计常有较大的差异。

此外，如对两台的记录作 Fourier 变换，分别求得 $\phi_2$ 及 $\phi_1$ 代入(8-2.19)式，所得的误差较大。应用互相关法求两台记录的相位差 $\phi_2-\phi_1$ 可以获得更好的结果。

设 $x_1(t)$、$x_2(t)$ 分别为两台的面波记录，它们的互相关函数为

$$\gamma_{21}(p)=\int_{-\infty}^{\infty}x_2(t+p)x_1(t)dt. \tag{8-2-20}$$

对求出的互相关函数作 Fourier 变换

$$R(f)=\int_{-\infty}^{\infty}\gamma_{21}(p)e^{-j2\pi fp}dp=X_2(f)X_1^*(f)$$

$$=|X_2(f)||X_1(f)|e^{j(\phi_2-\phi_1)}=|R(f)|e^{j\Delta\phi}. \tag{8-2-21}$$

式中 $X_1^*(f)$ 是 $X_1(f)$ 的共轭谱。所以互相关谱 $R(f)$ 的相位就是求相速度所需的相位差 $\Delta\phi=\phi_2-\phi_1$，代入(8-2.19)式，就可算出结果。图8-2-4为相位差法计算相速度的框图。

**2. 窄带通滤波——互相关函数法**

相位差法由于 $l$ 值的不确定，使相速度的计算有一定困难。因而提出改进的窄带通滤波——互相关函数法。这种方法不是基于相位对比，而是基于振幅对比。其主要步骤是先对面波信号作窄带通滤波，再计算互相关函数 $\gamma_{21}$，最后将归一化的 $\gamma_{21}$ 按矩阵排列形式输出，便可由等值线确定相速度 $c(T)$。

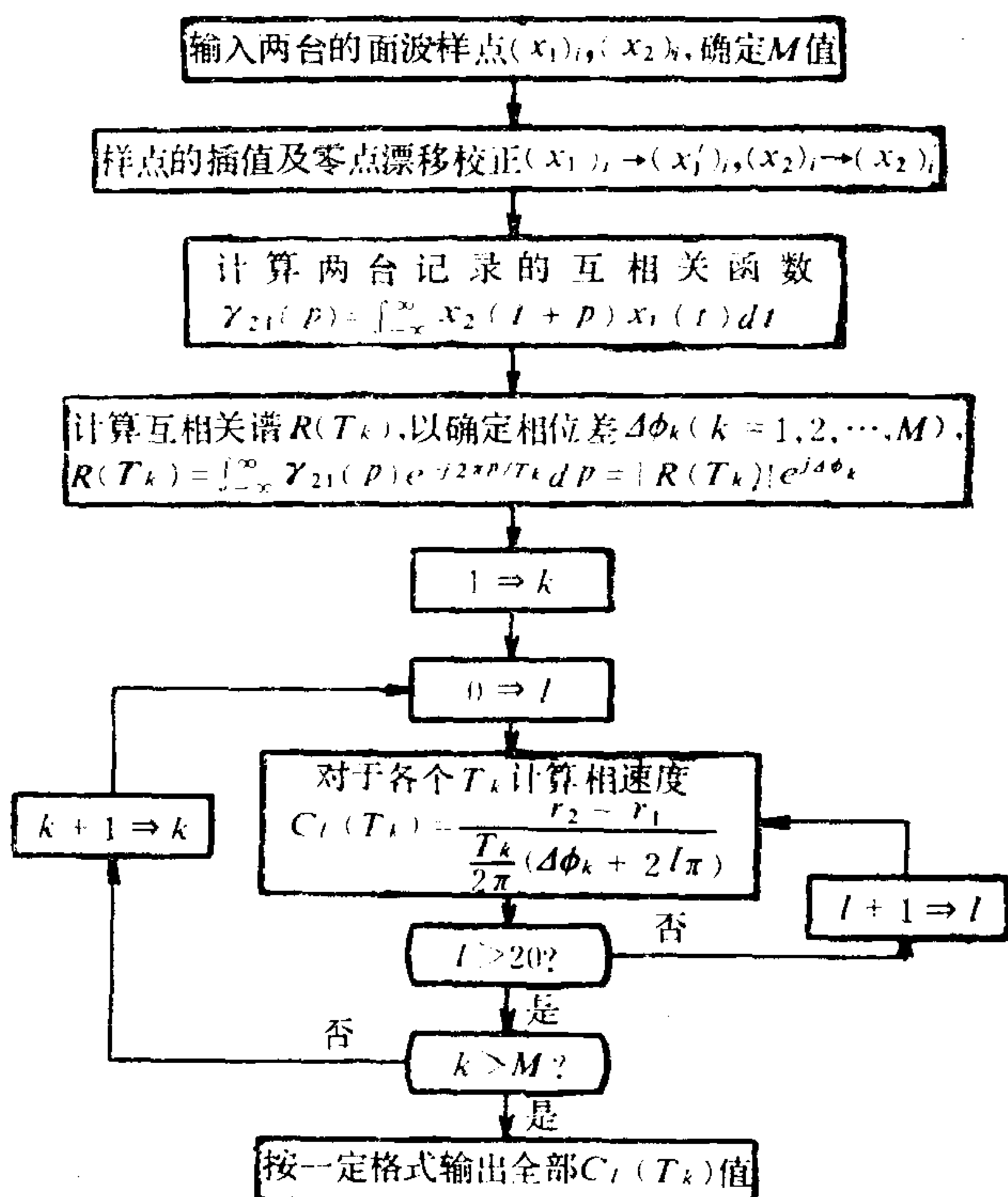



图 8-2-4

设 $x_1(t)$、$x_2(t)$ 为两台面波记录，首先分别对它们进行窄带通数字滤波，即在时间域内对这两个时间序列作褶积运算

$$y_1(\tau)=\int_{-\infty}^{\infty} x_1(\tau-t)W(t)dt,$$
$$y_2(\tau)=\int_{-\infty}^{\infty} x_2(\tau-t)W(t)dt \quad . \tag{8-2.22}$$

一个理想带通滤波器的单位脉冲响应函数为

$$W(T_k,\ t)=A_0\frac{\sin(2\pi ht)}{\pi t}\cos\left(\frac{2\pi}{T_k}t\right) \ . \tag{8-2.23}$$

其中，$h$ 为通频带半宽度（一般取 $h=0.001$），$T_k$ 为滤波器的中心周期，$A_0$ 为常系数。对有限长度的离散资料，为有效压制滤波器的边叶效应，必须对脉冲响应函数乘以一定的权函数 $u(t)$，$u(t)$ 可取半余弦平方函数。故窄带滤波器的脉冲响应函数为

$$W(T_k,\ t)=A_0\frac{\sin(2\pi ht)}{\pi t}\cos\left(\frac{2\pi}{T_k}t\right)\cos^2\left(\frac{\pi t}{10T_k}\right). \tag{8-2.24}$$

对两台分别作了窄带滤波后，下一步就是通过求互相关函数求相速度。因

$$\gamma_{21}(T_k,\ p)=\int_{-\infty}^{\infty}y_2(T_k,\ \tau+p)y_1(T_k,\tau)d\tau. \tag{8-2.25}$$

式中 $p$ 是第二台记录相对于第一台记录的延迟时间。若二台震中距为 $\Delta_2-\Delta_1$，则 $T_k$ 周期的相速度便为

$$c(T_k)=\frac{\Delta_2-\Delta_1}{p}. \tag{8-2.26}$$

因此，不妨将互相关函数记为

$$\gamma_{21}(T_k,\ p)=\gamma_{21}(T_k,\ c). \tag{8-2.27}$$

它表示 $T_k$ 周期在不同相速度 $c(T_k)$ 时的振幅分布。考虑到互相关函数有关系

$$\begin{aligned}\gamma_{21}(T_k,c)&=\frac{1}{2\pi}\int_{-\infty}^{\infty}Y_2(f)Y_1^*(f)e^{i2\pi f\tau}df\\&=\frac{1}{2\pi}\int_{-\infty}^{\infty}|Y_2(f)|\,|Y_1(f)|e^{i(\varphi_2-\varphi_1)}e^{i2\pi f\tau}df.\end{aligned} \tag{8-2.28}$$

由(8-2.28)式可以看出，当两 台 信 号 $y_1(t)$，$y_2(t)$ 同 相 时，$\phi_2-\phi_1=0$，$\gamma_{21}$有最大值。反相时，$\phi_2-\phi_1=\pm\pi$，$\gamma_{21}$为最小值。因此，将 $\gamma_{21}(T_k, c)$ 按 矩阵形式输出，将最大值等值线勾

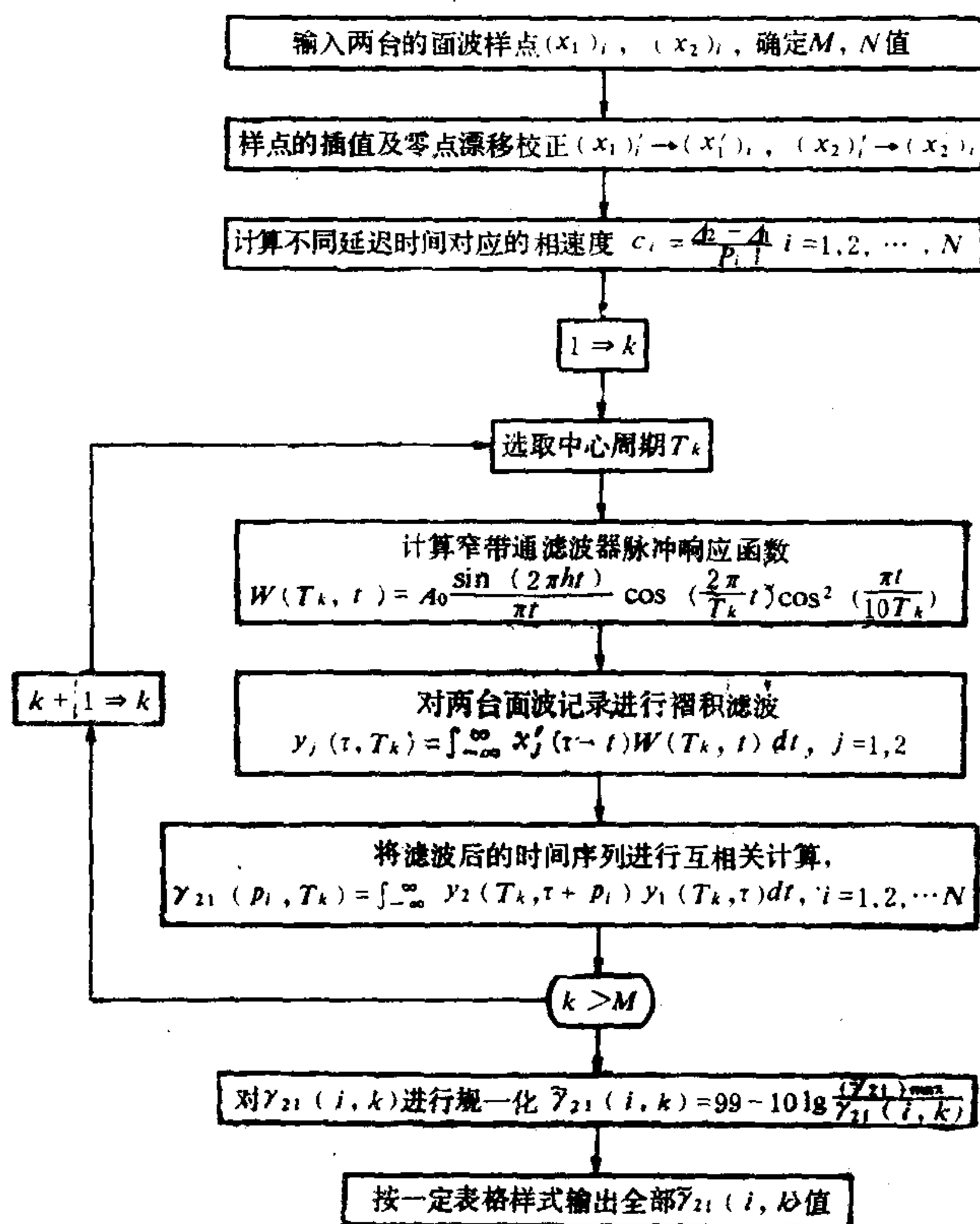



图 8-2-5

绘出来，即可以方便地确定相速度 频 散 曲 线 $c(T_k)$。输 出 $\gamma_{21}$ 时，仍可按最大振幅为 99 分贝进行规范。图8-2-5为计算框图。

图 8-2-6 是实际测定的例子，可以看到正负区域呈明显的条带分布。在任一同期处，相邻二个正值区差一个波长，正、负区相差半个波长。

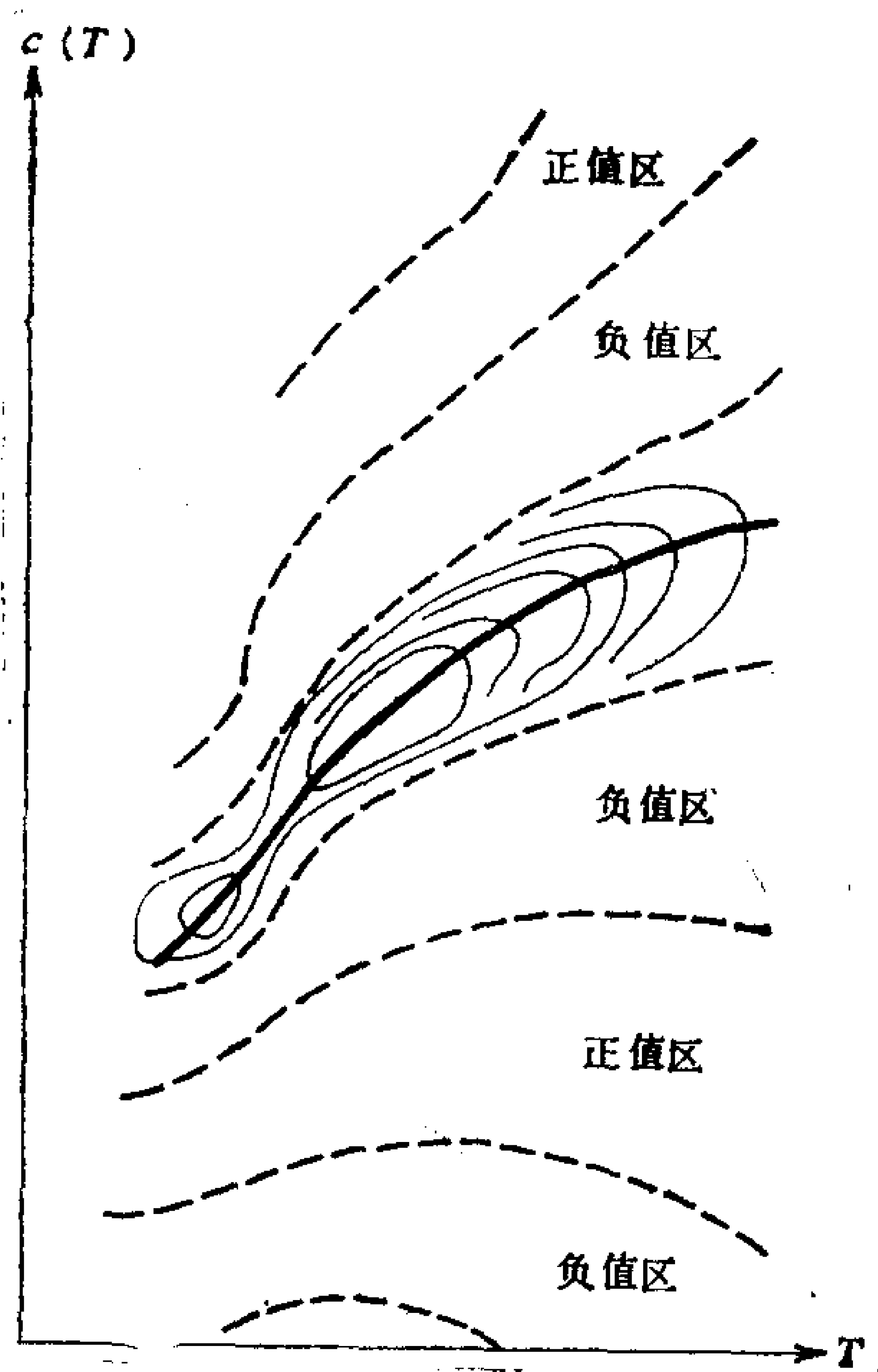



图 8-2-6

## § 8-3 多层介质面波频散计算的矩阵法

前面，我们已经讨论了 Rayleigh 波及 Love 波 的基本特性及频散曲线的测定方法。为研究地球内部结构，需要计算多层介质，而且包括有低速层、液体层(如海洋水层、液体外核)的各种情况，这就要求研究更一般的方法。

多层介质地震波的计算方法是从50年代发展起来的。首先是 Thomson 用矩阵方法解决体波计算，接着 Haskell 就 将它改进

并推广到面波问题的计算。Knopoff 于 1964 年讨论了混合的 Thomson-Haskell 方法，接着于1970 年在计算机上实现了这种算法。这一节，我们讨论 Haskell 矩阵法。

考虑圆频率为 $\omega$，相速度为 $c$ 的地震波在 $n$ 个水平的、均匀介质组成的层状空间中的传播问题。层状空间如图 8-3-1 所示。$x$ 轴与波的传播方向一致，$z$ 轴向下。层的序数及界面序数如图所示。

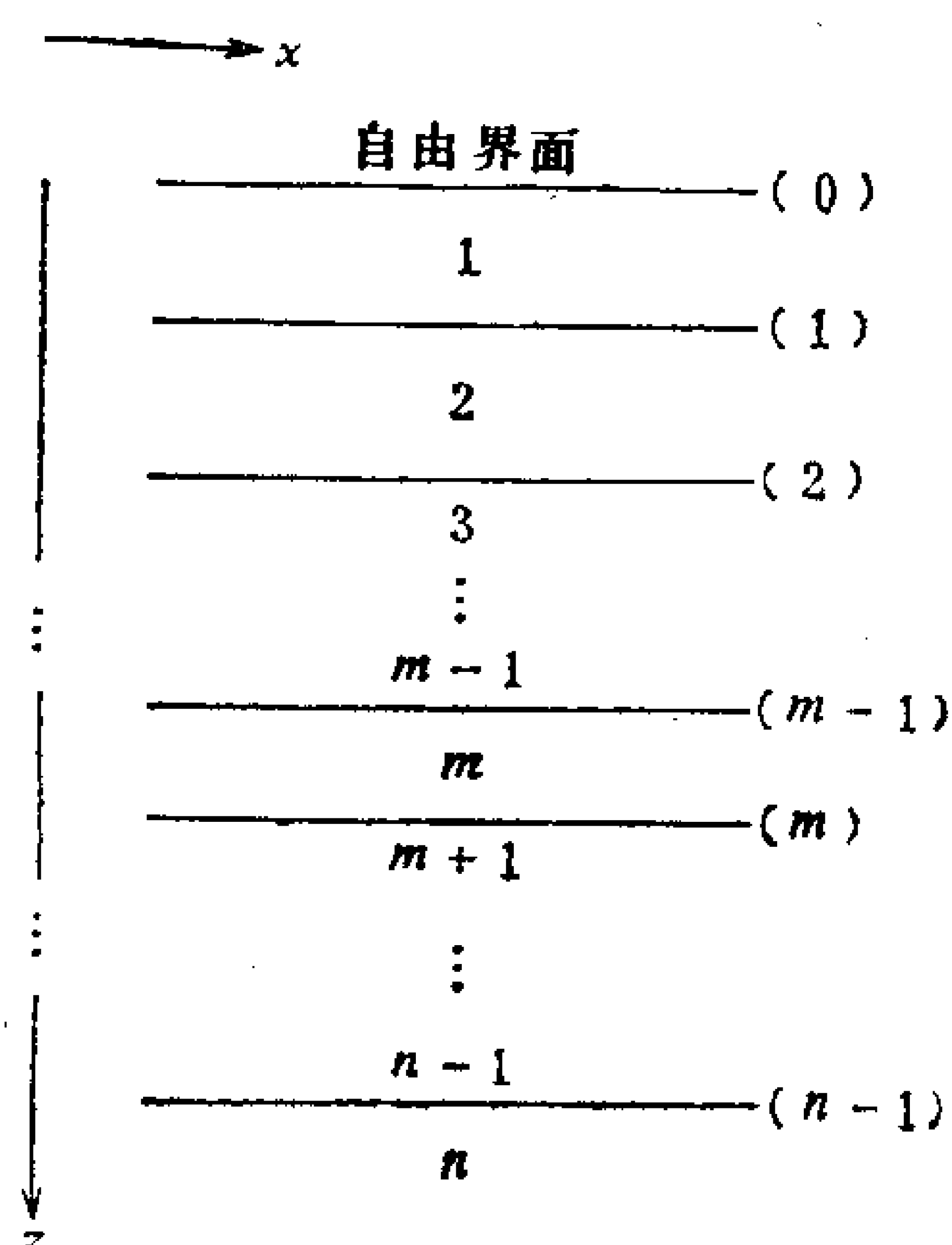


图 8-3-1

由于介质内部的不连续界面处存在四个边界条件（两个位移分量和两个应力分量连续），自由表面处有两个边界条件（两个应力分量为零）。所以，对 Rayleigh 波的计算总共有 $4n-2$个边界条件,这将导致$4n-2$个齐次联立方程式。仅当方程组系数组成的行列式为零时，此方程组有解。层数愈多，行列式的阶数就越高(例如，3 层为10阶，5 层为 18 阶)，计算越冗长。为了改进这种情况，就提出用迭代过程的矩阵方法。

这里我们先考虑 Rayleigh 波问题。据此，假定沿 $y$ 方向无位移且波的振幅随 $z$ 增大而衰减。对于第 $m$ 层，其层参数分别为：

$\rho_m$　密度；

$d_m$　厚度；

$\lambda_m$，$\mu_m$　Lame 常数；

$\alpha_m=[(\lambda_m+2\mu_m)/\rho_m]^{1/2}$　纵波波速；

$\beta_m=[\mu_m/\rho_m]^{1/2}$　横波波速；

$$r_{\alpha m}=\begin{cases}+[(c/\alpha_m)^2-1]^{1/2} & 当\ c>\alpha_m;\\ -i[1-(c/\alpha_m)^2]^{1/2} & 当\ c<\alpha_m;\end{cases}$$

$$r_{\beta m}=\begin{cases}+[(c/\beta_m)^2-1]^{1/2} & 当\ c>\beta_m,\\ -i[1-(c/\beta_m)^2]^{1/2} & 当\ c<\beta_m;\end{cases}$$

$\gamma_m=2(\beta_m/c)^2$;

$u$，$\omega$ 沿 $x$、$z$ 方向的位移；

$\delta=p_{zz}$　法向应力；

$\tau=p_{zx}$　切向应力；

$k=\dfrac{\omega}{c}=2\pi/\lambda$　波数。

众所周知，平面波在层状介质中传播，其波动方程为

$$\Delta^2\phi=\frac{1}{\alpha^2}\ \frac{\partial^2\phi}{\partial t^2},$$

$$\Delta^2\psi=\frac{1}{\beta^2}\ \frac{\partial^2\psi}{\partial t^2}. \tag{8-3.1}$$

式中，$\phi$、$\psi$ 分别表示纵波位和横波位。(8-3.1)式在第 $m$ 层中的解也可以直接表示成纵波和横波的组合。对于纵波有

$$\Delta_m=\frac{\partial u}{\partial x}+\frac{\partial w}{\partial z}=e^{j(\omega t-kx)}[\Delta'_m e^{-jkr_{\alpha m}z}+\Delta''_m e^{jkr_{\alpha m}z}] \tag{8-3.2}$$

对于横波有

$$\omega_m=\frac{1}{2}\left(\frac{\partial u}{\partial z}-\frac{\partial w}{\partial x}\right)=e^{j(\omega t-kx)}[\omega'_m e^{-jkr_{\beta m}z}+\omega''_m e^{jkr_{\beta m}z}]. \tag{8-3.3}$$

在(8-3.2)与(8-3.3)中，$\Delta_m$、$\omega_m$ 与纵波位及横波位的关系为

$$\phi=-\left(\frac{\alpha_m}{\omega}\right)^2\Delta_m,$$

$$\psi=2\left(\frac{\beta_m}{\omega}\right)^2\omega_m . \qquad (8\text{-}3.4)$$

而 $\Delta_m'$、$\Delta_m''$、$\omega_m'$、$\omega_m''$ 为常系数。其中 $\Delta_m'$ 的意义是：当 $r_{am}$ 为实数时，它表示与 $z$ 轴正方向夹角沿 $\mathrm{ctg}^{-1}r_{am}$ 方向传播的平面波，即通过层状介质传播的体波；当 $r_{am}$ 为虚数时，$\Delta_m'$ 为沿着 $x$ 轴正方向传播的面波，其振幅随深度增大而呈指数衰减。$\Delta_m''$ 的意义是，$r_{am}$ 为实数时，表示与 $z$ 轴负方向以上述同样夹角转播的体波；当 $r_{am}$ 为虚数时，亦为沿 $x$ 轴正方向传播的面波。对于 $\omega_m'$ 及 $\omega_m''$ 项，只要将 $r_{am}$ 改为 $r_{\beta m}$，也可按上述类似方式定义。

在第 $m$ 个界面的位移及应力分量为

$$u_m=-\left(\frac{\alpha_m}{\omega}\right)^2\frac{\partial\Delta_m}{\partial x}-2\left(\frac{\beta_m}{\omega}\right)^2\frac{\partial\omega_m}{\partial z},$$

$$w_m=-\left(\frac{\alpha_m}{\omega}\right)^2\frac{\partial\Delta_m}{\partial z}+2\left(\frac{\beta_m}{\omega}\right)^2\frac{\partial\omega_m}{\partial x},$$

$$(8\text{-}3.5)$$

$$\sigma_m=\rho_m\left\{\alpha_m^2\Delta_m+2\beta_m^2\left[\left(\frac{\alpha_m}{\omega}\right)^2\frac{\partial^2\Delta_m}{\partial x^2}+2\left(\frac{\beta_m}{\omega}\right)^2\frac{\partial^2\omega_m}{\partial x\partial z}\right]\right\},$$

$$\tau_m=2\rho_m\beta_m^2\left\{-\left(\frac{\alpha_m}{\omega}\right)^2\frac{\partial^2\Delta_m}{\partial x\partial z}+\left(\frac{\beta_m}{\omega}\right)^2\left[\frac{\partial^2\omega_m}{\partial x^2}-\frac{\partial^2\omega_m}{\partial z^2}\right]\right\}.$$

在层状介质内任一界面上，这四个量都是连续的。位移的连续性使相应的速度分量 $\dot{u}$ 及 $\dot{w}$ 也连续，由于相速度 $c$ 对各层都一样，因此，无量纲的量 $\dot{u}/c$ 及 $\dot{w}/c$ 也连续。将(8-3.2)及(8-3.3)式代入(8-3.5)式，并将含有 $jkrz$ 因子的指数函数改为三角函数，可得

$$\begin{aligned}\dot{u}_m/c=&-(\alpha_m/c)^2[(\Delta_m'+\Delta_m'')\cos kr_{am}z-j(\Delta_m'-\Delta_m'')\\&\times\sin kr_{am}z]-\gamma_m r_{\beta m}[(\omega_m'-\omega_m'')\cos kr_{\beta m}z-j(\omega_m'\\&+\omega_m'')\sin kr_{\beta m}z],\end{aligned}$$

$$\dot{w}_m/c=-(\alpha_m/c)^2 r_{\alpha m}[-j(\Delta_m'+\Delta_m'')\sin kr_{\alpha m}z+(\Delta_m'-\Delta_m'')\cos kr_{\alpha m}z]+\gamma_m[-j(\omega_m'-\omega_m'')\sin kr_{\beta m}z+(\omega_m'+\omega_m'')\cos kr_{\beta m}z],$$

$$\sigma_m=-\rho_m\alpha_m^2(\gamma_m-1)[(\Delta_m'+\Delta_m'')\cos kr_{\alpha m}z-j(\Delta_m'-\Delta_m'')\sin kr_{\alpha m}z]-\rho_m c^2\gamma^2{}_m\gamma_{\beta m}[(\omega_m'-\omega_m'')\cos kr_{\beta m}z-j(\omega_m'+\omega_m'')\sin kr_{\beta m}z],$$

$$\tau_m=\rho_m\alpha_m^2\gamma_m r_{\alpha m}[-j(\Delta_m'+\Delta_m'')\sin kr_{\alpha m}z+(\Delta_m'-\Delta_m'')\cos kr_{\alpha m}z]-\rho_m c^2\gamma_m(\gamma_m-1)[-j(\omega_m'-\omega_m'')\sin kr_{\beta m}z+(\omega_m'+\omega_m'')\cos kr_{\beta m}z]。$$

(8-3.6)

将 $z$ 轴的原点置于 $(m\text{-}1)$ 界面上，则在 $(m\text{-}1)$ 界面 处 $z=0$，于是 $\sin krz=0$，$\cos krz=1$。(8-3.6)式中 $\dot{u}/c$，$\dot{w}/c$，$\sigma$ 与 $\tau$ 在 $(m\text{-}1)$ 界面上的值，可表示 为 常 数 $(\Delta_m'+\Delta_m'')$，$(\Delta_m'-\Delta_m'')$，$(\omega_m'-\omega_m'')$及 $(\omega_m'+\omega_m'')$ 的下列线性变换关系

$$\left(\frac{\dot{u}_{m-1}}{c},\ \frac{\dot{w}_{m-1}}{c},\ \sigma_{m-1},\ \tau_{m-1}\right)^T=E_m(\Delta_m'+\Delta_m'',\ \Delta_m'-\Delta_m'',\ \omega_m'-\omega_m'',\ \omega_m'+\omega_m'')^T. \qquad (8\text{-}3.7)$$

这里矩阵 $E_m$ 不难由(8-3.6)式确定，

$$E_m=\begin{pmatrix} -\left(\frac{\alpha_m}{c}\right)^2 & 0 & -\gamma_m r_{\beta m} & 0 \\ 0 & -\left(\frac{\alpha_m}{c}\right)^2 r_{\alpha m} & 0 & \gamma_m \\ -\rho_m\alpha_m^2(\gamma_m-1) & 0 & -\rho_m c^2\gamma_m^2 r_{\beta m} & 0 \\ 0 & \rho_m\alpha_m^2\gamma_m r_{\alpha m} & 0 & -\rho_m c^2\gamma_m(\gamma_m-1) \end{pmatrix},$$

(8-3.8)

用同样的办法，将 $z$ 轴原点置于 $(m)$ 界面 上， 即 取 $z=d_m$ 代入

(8-3.6)式，可得下列线性变换关系

$$\left(\frac{\dot{u}_m}{c},\ \frac{\dot{w}_m}{c},\ \sigma_m,\ \tau_m\right)^T = D_m(\Delta_m' + \Delta_m'', \Delta_m' - \Delta_m'', \omega_m' - \omega_m'',\ \omega_m' + \omega_m'')^T. \tag{8-3.9}$$

$D_m$ 矩阵为

$$D_m = \begin{pmatrix} -(\alpha_m/c)^2\cos P_m & j(\alpha_m/c)^2\sin P_m & -\gamma_m r_{\beta m}\cos Q_m & j\gamma_m r_{\beta m}\sin Q_m \\ j(\alpha_m/c)^2 r_{\alpha m}\sin P_m & -(\alpha_m/c)^2 r_{\alpha m}\cos P_m & -j\gamma_m\sin Q_m & \gamma_m\cos Q_m \\ -\rho_m\alpha_m^2(\gamma_m-1)\cos P_m & j\rho_m\alpha_m^2(\gamma_m-1)\sin P_m & -\rho_m c^2\gamma_m^2 r_{\beta m}\cos Q_m & j\rho_m c^2\gamma_m^2 r_{\beta m}\sin Q_m \\ -j\rho_m\alpha_m^2\gamma_m r_{\alpha m}\sin P_m & P_m\alpha_m^2\gamma_m r_{\alpha m}\cos P_m & j\rho_m c^2\gamma_m(\gamma_m-1)\sin Q_m & -P_m c^2\gamma_m(\gamma_m-1)\cos Q_m \end{pmatrix}. \tag{8-3.10}$$

其中 $P_m = kr_{\alpha m}d_m$，$Q_m = kr_{\beta m}d_m$。

由(8-3.7)式得

$$(\Delta_m' + \Delta_m'', \Delta_m' - \Delta_m'', \omega_m' - \omega_m'', \omega_m' + \omega_m'')^T = E_m^{-1}\left(\frac{\dot{u}_{m-1}}{c}, \frac{\dot{w}_{m-1}}{c}, \sigma_{m-1}, \tau_{m-1}\right)^T. \tag{8-3.11}$$

代入(8-3.9)式得

$$\left(\frac{\dot{u}_m}{c},\ \frac{\dot{w}_m}{c},\ \sigma_m,\ \tau_m\right)^T = D_m E_m^{-1}\left(\frac{\dot{u}_{m-1}}{c},\ \frac{\dot{w}_{m-1}}{c},\ \sigma_{m-1},\ \tau_{m-1}\right)^T. \tag{8-3.12}$$

其中

$$
E_m^{-1}=\begin{pmatrix} -2(\beta_m/\alpha_m)^2 & 0 & (\rho_m\alpha_m^2)^{-1} & 0 \\ 0 & c^2(\gamma_m-1)/\alpha_m^2 r_{\alpha m} & 0 & (\rho_m\alpha_m^2 r_{\alpha m})^{-1} \\ (\gamma_m-1)/\gamma_m r_{\beta m} & 0 & -(\rho_m c^2\gamma_m r_{\beta m})^{-1} & 0 \\ 0 & 1 & 0 & (\rho_m c^2\gamma_m)^{-1} \end{pmatrix}. \tag{8-3.13}
$$

取(8-3.12)式中矩阵的积为

$$
a_m=D_m E_m^{-1}.
$$

由(8-3.10)及(8-3.13)式，可得 $a_m$ 的元素为

$$
\begin{aligned}
&(a_m)_{11}=\gamma_m\cos P_m-(\gamma_m-1)\cos Q_m,\\
&(a_m)_{12}=j[(\gamma_m-1)\gamma_{\alpha m}^{-1}\sin P_m+\gamma_m r_{\beta m}\sin Q_m],\\
&(a_m)_{13}=-(\rho_m c^2)^{-1}(\cos p_m-\cos Q_m),\\
&(a_m)_{14}=j(\rho_m c^2)^{-1}(\gamma_{\alpha m}^{-1}\sin P_m+r_{\beta m}\sin Q_m),\\
&(a_m)_{21}=-j[\gamma_m r_{\alpha m}\sin P_m+(\gamma_m-1)\gamma_{\beta m}^{-1}\sin Q_m],\\
&(a_m)_{22}=-(\gamma_m-1)\cos P_m+\gamma_m\cos Q_m,\\
&(a_m)_{23}=j(\rho_m c^2)^{-1}(r_{\alpha m}\sin P_m+\gamma_{\beta m}^{-1}\sin Q_m),\\
&(a_m)_{24}=(a_m)_{13},\\
&(a_m)_{31}=\rho_m c^2\gamma_m(\gamma_m-1)(\cos P_m-\cos Q_m),\\
&(a_m)_{32}=j\rho_m c^2[(\gamma_m-1)^2\gamma_{\alpha m}^{-1}\sin p_m+\gamma_m^2 r_{\beta m}\sin Q_m],\\
&(a_m)_{33}=(a_m)_{22},\\
&(a_m)_{34}=(a_m)_{12},\\
&(a_m)_{41}=j\rho_m c^2[\gamma_m^2 r_{\alpha m}\sin P_m+(\gamma_m-1)^2\gamma_{\beta m}^{-1}\sin Q_m],\\
&(a_m)_{42}=(a_m)_{31},\\
&(a_m)_{43}=(a_m)_{21},\\
&(a_m)_{44}=(a_m)_{11}.
\end{aligned} \tag{8-3.14}
$$

显见 $a_m$ 是一个反对称矩阵，即有下列形式

$$
\boldsymbol{a}_m=\begin{pmatrix}
(a_m)_{11} & (a_m)_{12} & (a_m)_{13} & (a_m)_{14}\\
(a_m)_{21} & (a_m)_{22} & (a_m)_{23} & (a_m)_{13}\\
(a_m)_{31} & (a_m)_{32} & (a_m)_{22} & (a_m)_{12}\\
(a_n)_{41} & (a_m)_{31} & (a_m)_{21} & (a_m)_{11}
\end{pmatrix}. \tag{8-3.15}
$$

因而(8-3.12)式可以写为

$$
\left(\frac{\dot{u}_m}{c},\ \frac{\dot{w}_m}{c},\ \sigma_m,\ \dot{\tau}_m\right)^T=a_m\left(\frac{\dot{u}_{m-1}}{c},\ \frac{\dot{w}_{m-1}}{c},\ \sigma_{m-1},\ \tau_{m-1}\right)^T
$$

$$
\tag{8-3.16}
$$

上式中以$(m-1)$代$(m)$得到

$$
\left(\frac{\dot{u}_{m-1}}{c},\ \frac{\dot{w}_{m-1}}{c},\ \sigma_{m-1},\ \tau_{m-1}\right)^T=a_{m-1}\left(\frac{\dot{u}_{m-2}}{c},\ \frac{\dot{w}_{m-2}}{c},\right.
$$

$$
\left.\sigma_{m-2},\ \tau_{m-2}\right)^T. \tag{8-3.17}
$$

将(8-3.17)代入(8-3.16)式得

$$
\left(\frac{\dot{u}_m}{c},\ \frac{\dot{w}_m}{c},\ \sigma_m,\ \tau_m\right)^T=\mathbf{a}_m\mathbf{a}_{m-1}\left(\frac{\dot{u}_{m-2}}{c},\ \frac{\dot{w}_{m-2}}{c},\right.
$$

$$
\left.\sigma_{m-2},\ \tau_{m-2}\right). \tag{8-3.18}
$$

反复利用这个过程可以得到

$$
\left(\frac{\dot{u}_{n-1}}{c},\ \frac{\dot{w}_{n-1}}{c},\ \sigma_{n-1},\ \tau_{n-1}\right)
$$

$$
=a_{n-1}a_{n-2}\cdots a_1\left(\frac{\dot{u}_0}{c},\ \frac{\dot{w}_0}{c},\ \sigma_0,\ \tau_0\right)^T.
$$

$$
\tag{8-3.19}
$$

对于第 $n$ 层，由(8-3.11)式得

$$(\Delta'_n+\Delta'_n, \Delta'_n-\Delta''_n,\ \omega'_n-\omega''_n,\ \omega'_n+\omega''_n)$$

$$=E^{-1}_n\left(\frac{\dot{u}_{n-1}}{c},\ \frac{\dot{w}_{n-1}}{c},\ \sigma_{n-1},\ \tau_{n-1}\right)^T.$$

(8-3.20)

将(8-3.19)代入上式得

$$(\Delta'_n+\Delta''_n,\ \Delta'_n-\Delta''_n,\ \omega'_n-\omega''_n,\ \omega'_n+\omega''_n)^T$$

$$=E^{-1}_n\boldsymbol{a}_{n-1}\boldsymbol{a}_{n-2}\cdots\boldsymbol{a}_2\boldsymbol{a}_1\cdot\left(\frac{\dot{u}_0}{c},\ \frac{\dot{w}_0}{c},\ \sigma_0,\ \tau_0\right)^T.$$

(8-3.21)

这是一个通用形式,对多层介质中的体波和面波都成立。对于我们研究目的来说,我们特别要讨论面波的情况,即在自由表面处应力为零，因此 $\sigma_0=\tau_0=0$；在无穷远处无震源，因而 $\Delta''_n=\omega''_n=0$，并且令

$$J=E^{-1}_n\boldsymbol{a}_{n-1}\boldsymbol{a}_{n-2}\cdots\boldsymbol{a}_2\boldsymbol{a}_1.$$

于是(8-3.21)式便写为

$$(\Delta_n',\ \Delta_n',\ \omega_n',\ \omega_n')^T=J\left(\frac{\dot{u}_0}{c},\ \frac{\dot{w}_0}{c},\ 0,\ 0\right)^T$$

(8-3.22)

其中 $J$ 为 $4\times4$ 方阵，将(8-3.22)展开得到

$$\Delta_n'=J_{11}\frac{\dot{u}_0}{c}+J_{12}\frac{\dot{w}_0}{c},$$

$$\Delta_n'=J_{21}\frac{\dot{u}_0}{c}+J_{22}\frac{\dot{w}_0}{c},$$

$$\omega_n'=J_{31}\frac{\dot{u}_0}{c}+J_{32}\frac{\dot{w}_0}{c},$$

$$\omega_n' = J_{41}\frac{\dot{u}_0}{c} + J_{42}\frac{\dot{w}_0}{c} \ . \tag{8-3.23}$$

由上式消去 $\Delta_n'$及 $\omega_n'$得

$$\frac{\dot{u}_0}{\dot{w}_0} = \frac{J_{22} - J_{12}}{J_{11} - J_{21}} = \frac{J_{42} - J_{32}}{J_{31} - J_{41}} \ . \tag{8-3.24}$$

由于矩阵 $J$ 的元素是相速度 $c$ 和波数 $k$ 的函 数，所 以 方程(8-3.24)式提供了 $c$ 与 $k$ 的隐含函数关系，这即是所求的相速度 $c$ 的频散函数。

下面进一步讨论频散函数的一般特性。令

$$A = a_{n-1}a_{n-2}\cdots a_2 a_1 .$$

它是一个4 × 4 矩阵，于是 $J$ 矩阵可写为

$$J = E_n^{-1}A . \tag{8-3.25}$$

将(8-3.24)改写为以下形式

$$-\frac{\dot{u}_0}{\dot{w}_0} = \frac{K}{L} = \frac{M}{N} \ . \tag{8-3.26}$$

其中各系数可以由(8-3.25)算出

$$K = \gamma_n r_{\alpha n} A_{12} + (\gamma_n - 1)A_{22} - r_{\alpha n}A_{32}/\rho_n c^2 + A_{42}/\rho_n c^2,$$

$$L = \gamma_n r_{\alpha n} A_{11} + (\gamma_n - 1)A_{21} - r_{\alpha n}A_{31}/\rho_n c^2 + A_{41}/\rho_n c^2,$$

$$M = (\gamma_n - 1)A_{12} + \gamma_n r_{\beta n}A_{22} + A_{32}/\rho_n c^2 + r_{\beta n}A_{42}/\rho_n c^2,$$

$$N = -(\gamma_n - 1)A_{11} + \gamma_n r_{\beta n}A_{21} + A_{31}/\rho_n c^2 + r_{\beta n}A_{41}/\rho_n c^2 \ . \tag{8-3.27}$$

由前述可知，矩阵 $a_m$ 的元 素 是由 $\sin P_m$，$\sin Q_m$、$r_{\beta m}$ 及

$r_{\alpha m}$ 等量确定的，而这些量是实数还是虚数取决于相速度 $c$ 的大小，而且它们仅以 $r^{\pm}_{\alpha m}\sin P_m$ 及 $r^{\pm}_{\beta m}\sin Q_m$ 的组合形式出现。因为 $\sin P_m$ 是实数还是虚数是和 $r_{\alpha m}$ 一致的，而 $\sin Q_m$ 也与 $r_{\beta m}$ 一致。所以，只要相速度为实数，上述组合形式总是实的。按照这种特性，矩阵 $a_m$ 的元素具有下述形式：

$$\mathbf{a}_m=\begin{pmatrix} R & I & R & I \\ I & R & I & R \\ R & I & R & I \\ I & R & I & R \end{pmatrix}. \tag{8-3.28}$$

式中 $R$ 表示实数，$I$ 表示虚数。因此积矩阵 $A=\mathbf{a}_{n-1}\mathbf{a}_{n-2}\cdots\mathbf{a}_2\mathbf{a}_1$，也具有同样的实数虚数分布形式。即(8-3.27)式中 $A_{11}$，$A_{22}$，$A_{31}$ 及 $A_{42}$ 为实数；$A_{12}$，$A_{21}$，$A_{32}$ 及 $A_{41}$ 为虚数。

由于我们考虑面波情况，即相速度 $c<\beta_n$，故 $r_{\alpha n}$ 及 $r_{\beta n}$ 必然为虚数。由 (8-3.27) 知，$K$，$N$ 中所有各项都为实数，而 $L$，$M$ 中所有各项都是虚数。于是(8-3.26)中比值 $\frac{\dot{u}_0}{\dot{\omega}_0}$ 总是纯虚数。这表明自由表面处垂直方向位移与水平方向位移有90℃相位差。在有介质耗散的情况下，波数 $k$ 为复数，$\frac{\dot{u}_0}{\dot{w}_0}$ 是复数而不再是纯虚数，垂直位移与水平位移相位差不等于 90°，极化椭圆轴是倾斜的。

## §8-4　多层介质Rayleigh波频散快速计算法

由 Haskell 发展的矩阵方法虽然对多层介质面波频散计算较过去有很大改进，但迭代过程中仍存在有效数字损失和计算速度较低的缺点。这一节讨论 Knopff 提出的改进模型，这是目前使

用的较有效的计算机快速计算方法。

多层介质的模型及坐标系选取时，要考虑到各种可能的情况，如图8-4-1示。

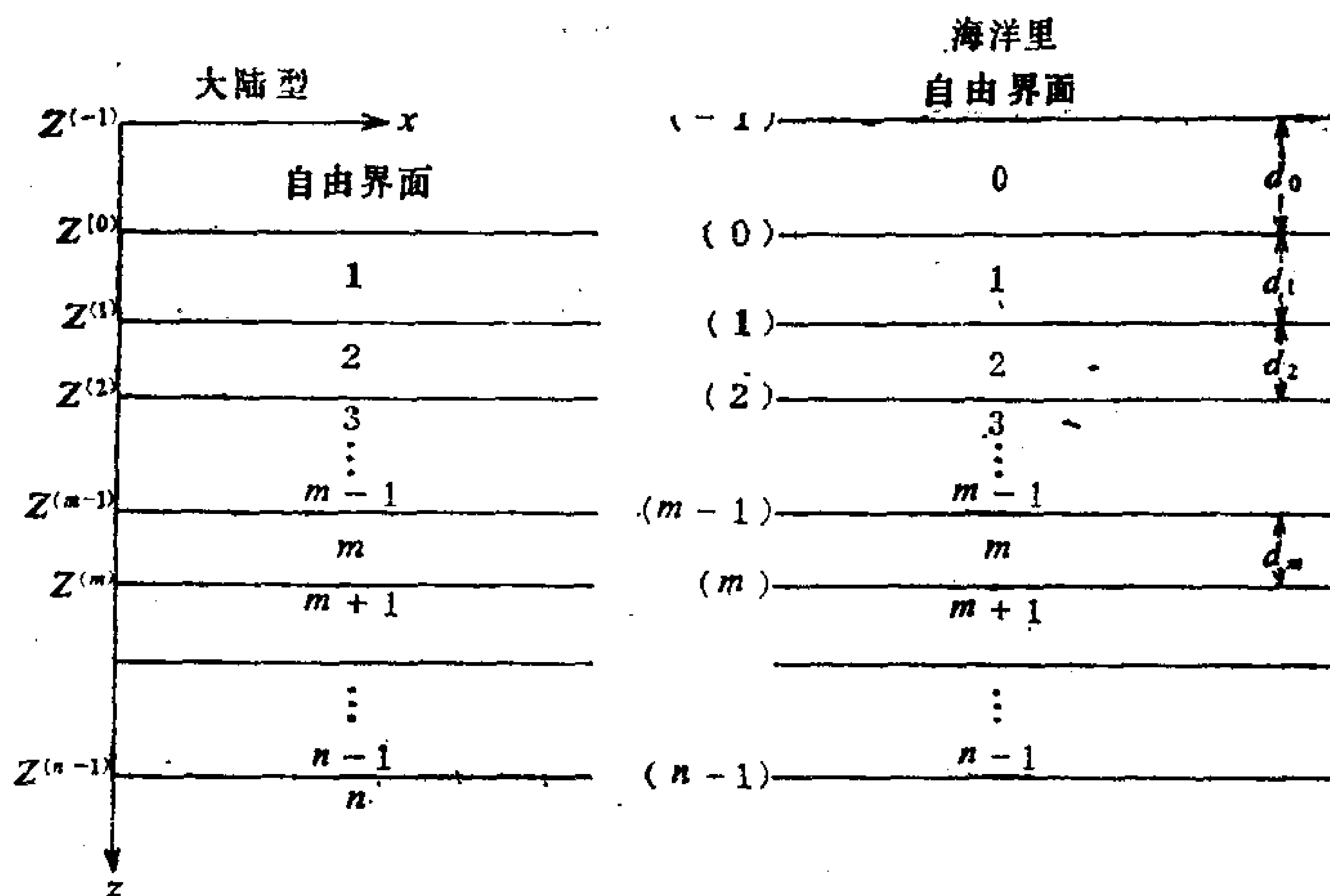



图 8-4-1

图中左侧表示 $n$-1 层完全弹性均匀介质覆盖在半空间上，这相当于大陆型结构；右侧在最上面还覆盖有液体层，这相当于海洋型结构。每种模型根据其结构又分为三种情形：即最深界面在地幔中的固体-固体模型；最深界面在外核中的固体-液体模型；最深界面在内核中的固体-液体-固体模型。下面分别讨论。

## （一） 最深界面在地幔中的固体–固体模型

上一节中，我们已经根据Haskell方法推导出位移与应力的表达式(8-3.6)，现在对它作如下变换：在$\frac{\dot{u}_m}{c}$及$\frac{\dot{w}_m}{c}$两端同乘以 $c^2$；以 $z-z_{m-1}$代替 $Z$，令

$$A_m=-\alpha_m^2(\Delta_m'+\Delta_m''),$$

$$B_m=-\alpha_m^2(\Delta_m'-\Delta_m''),$$

$$C_m = -2\beta_m^2(\omega_m' - \omega_m^*),$$

$$D_m = -2\beta_m^2(\omega_m' + \omega_m^*),$$

$$p_m = kr_{am}(z - z_m),$$

$$Q_m = kr_{\beta m}(z - z_m).$$

并注意到 $\gamma_m = 2\left(\frac{\beta_m}{c}\right)^2$，于是(8-3.6)式可以写为

$$\begin{aligned}
c\dot{u}_m &= \cos P_m A_m + j\sin P_m B_m + r_{\beta m}\cos Q_m C_m - jr_{\beta m}\sin Q_m D_m,\\
c\dot{w}_m &= -jr_{am}\sin P_m A_m + r_{am}\cos P_m B_m + j\sin Q_m C_m\\
&\quad - \cos Q_m D_m,\\
\sigma_m &= \rho_m(\gamma_{m-1})\cos P_m A_m - j\rho_m(\gamma_m - 1)\sin P_m B_m\\
&\quad + \rho_m\gamma_m r_{\beta m}\cos Q_m C_m - j\rho_m\gamma_m r_{\beta m}\sin Q_m D_m,\\
\tau_m &= j\rho_m\gamma_m r_{am}\sin P_m A_m - \rho_m\gamma_m r_{am}\cos P_m B_m - j\rho_m(\gamma_m\\
&\quad - 1)\sin Q_m C_m + \rho_m(\gamma_m - 1)\cos Q_m D_m.
\end{aligned}$$

(8-4.1)

对于大陆模型，因自由表面上的应力为零，故上式中$\sigma_1 = \tau_1 = 0$，且由于 $z - z_0 = 0$，故 $P_1 = Q_1 = 0$，式(8-4.1)中后二式可简化为

$$-\rho_1(\gamma_1 - 1)A_1 - \rho_1\gamma_1 r_{\beta 1}C_1 = 0,$$

$$\rho_1\gamma_1 r_{a1}B_1 - \rho_1(\gamma_1 - 1)D_1 = 0.$$

(8-4.2)

上式可以写为矩阵形式，如取 $(A_1, B_1, C_1, D_1)^T$ 为向量，则系数可以写为 2 × 4 子矩阵形式：

$$A_{con}^{(0)} = \begin{pmatrix} -\rho_1(\gamma_1 - 1) & 0 & -\rho_1\gamma_1 r_{\beta 1} & 0 \\ 0 & \rho_1\gamma_1 r_{a1} & 0 & -\rho_1(\gamma_1 - 1) \end{pmatrix}.$$

(8-4.3)

对于海洋模型，自由表面（水面）$\sigma_0=0$，水层下 $z^{(0)}$ 界面位移及应力垂直分量连续，切向量为零的条件，可得以下 2 × 4 子矩阵：

$$\Lambda_{oce}^{(0)}=\begin{pmatrix} -\rho_1(\gamma_1-1) & j\rho_0 r_{\alpha 1}\mathrm{tg}P_0/r_{\alpha 0} & -\rho_1\gamma_1 r_{\beta 1} & -j\rho_0\mathrm{tg}p_0/r_{\alpha 0} \\ 0 & \rho_1\gamma_1 r_{\alpha 1} & 0 & -\rho_1(\gamma_1-1) \end{pmatrix}. \tag{8-4.4}$$

式中 $P_0=kr_{\alpha 0}d_0$。在（8-4.3）及（8-4.4）中下角标 *con* 表示大陆，*oce* 表示海洋。

对于内部任意 $m$ 界面，根据界面上位移及应力的连续条件，可以得到 4 × 8 界面子矩阵：

$$\Lambda^{(m)}=\begin{pmatrix} \cos P_m & -j\sin P_m & r_{\beta m}\cos Q_m & -jr_{\beta m}\sin Q_m & -1 & 0 & -r_{\beta m+1} & 0 \\ -jr_{\alpha m}\sin P_m & r_{\alpha m}\cos P_m & j\sin Q_m & -\cos Q_m & 0 & -r_{\alpha m+1} & 0 & 1 \\ \rho_m(\gamma_m-1)\cos P_m & -j\rho_m(\gamma_m-1)\sin P_m & \rho_m\gamma_m r_{\beta m}\cos Q_m & -j\rho_m\gamma_m r_{\beta m}\sin Q_m & -\rho_{m+1}(\gamma_{m+1}-1) & 0 & -\rho_{m+1}r_{\beta m+1}\gamma_{m+1} & 0 \\ -j\rho_m\gamma_m r_{\alpha m}\sin P_m & -\rho_m\gamma_m r_{\beta m}\cos P_m & -j\rho_m(\gamma_m-1)\sin Q_m & \rho_m(\gamma_m-1)\cos Q_m & 0 & \rho_{m+1}\gamma_{m+1}r_{m+1} & 0 & -\rho_{m+1}(\gamma_{m+1}-1) \end{pmatrix}. \tag{8-4.5}$$

式中 $P_m=kr_{\alpha m}d_m$，$Q_m=kr_{\beta m}d_m$。

注意到在第 $n$ 层 $\Delta_n''=\omega_n''=0$，所以 $A_n=B_n$，$C_n=D_n$，得到（$n$-1）界面子矩阵为 4 × 6 形式：

$$
\Lambda^{(n-1)}=\begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & -1 & -r_{\beta n} \\
\cdot & \cdot & \cdot & \cdot & -r_{an} & 1 \\
\cdot & \cdot & \cdot & \cdot & -\rho_n(\gamma_n-1) & -\rho_n\gamma_n r_{\beta n} \\
\cdot & \cdot & \cdot & \cdot & \rho_n\gamma_n r_{an} & -\rho_n(\gamma_n-1)
\end{pmatrix}. \tag{8-4.6}
$$

$\Lambda^{(n-1)}$ 中前四列元素与$\Lambda^{(m)}$的前四列形式完全相同，只须将有关下角标 $m$ 改为 $n-1$ 即可。

实用中还需对(8-4.3)—(8-4.6)各子矩阵进行整理。用 $r_{ai}$ 去除各子矩阵的 2 或 6 列，用 $r_{\beta i}$ 去除各子矩阵的 3 或 7 列（$i=1,2,\cdots\cdots,n-1$），则有以下结果：

$$
\Lambda^{(0)}_{con}=\begin{pmatrix}
-\rho_1(\gamma_1-1) & 0 & -\rho_1\gamma_1 & 0 \\
0 & \rho_1\gamma_1 & 0 & -\rho_1(\gamma_1-1)
\end{pmatrix}, \tag{8-4.7}
$$

$$
\Lambda^{(0)}_{oce}=\begin{pmatrix}
-\rho_1(\gamma_1-1) & j\rho_0\text{tg}P_0/r_{a0} & -\rho_1\gamma_1 & -j\rho_0\text{tg}P_0/r_{a0} \\
0 & \rho_1\gamma_1 & 0 & -\rho_1(\gamma_1-1)
\end{pmatrix}, \tag{8-4.8}
$$

$$
\Lambda^{(m)}=\begin{pmatrix}
\cos P_m & -j\sin P_m/r_{am} \\
-jr_{am}\sin P_m & \cos P_m \\
\rho_m(\gamma_m-1)\cos P_m & -j\rho_m(\gamma_m-1)\sin P_m/r_{am} \\
j\rho_m\gamma_m r_{am}\sin P_m & -\rho_m\gamma_m\cos P_m
\end{pmatrix.}
$$

$$
\begin{matrix}
\cos Q_m & jr_{\beta m}\sin Q_m \\
j\sin Q_m/r_{\beta m} & -\cos Q_m \\
\rho_m\gamma_m\cos Q_m & -j\rho_m\gamma_m r_{\beta m}\sin Q_m \\
-j\rho_m(\gamma_m-1)\sin Q_m/r_{\beta m} & \rho_m(\gamma_m-1)\cos Q_m
\end{matrix}
$$

$$
\begin{pmatrix}
-1 & 0 & -1 & 0 \\
0 & -1 & 0 & 1 \\
-\rho_{m+1}(\gamma_{m+1}-1) & 0 & -\rho_{m+1}\gamma_{m+1} & 0 \\
0 & \rho_{m+1}\gamma_{m+1} & 0 & -\rho_{m+1}(\gamma_{m+1}-1)
\end{pmatrix},
\tag{8-4.9}
$$

$$
A^{(n-1)} = \begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & -1 & -r_{\beta n} \\
\cdot & \cdot & \cdot & \cdot & -r_{\alpha n} & 1 \\
\cdot & \cdot & \cdot & \cdot & -\rho_n(\gamma_n-1) & -\rho_n\gamma_n r_{\beta n} \\
\cdot & \cdot & \cdot & \cdot & \rho_n\gamma_n r_{\alpha n} & -\rho_n(\gamma_n-1)
\end{pmatrix}.
\tag{8-4.10}
$$

$A^{(n-1)}$的前四列元素与$A^{(m)}$形式相同。将这些子矩阵组合起来，就构成下列频散函数的行列式形式：

$$
F_R(\omega,c) = \begin{pmatrix} \ddots & & \\ & \cdots & \\ & & \ddots \end{pmatrix}.
\tag{8-4.11}
$$

这是一个$(4n-2)$阶行列式，直接计算是困难的。因此必须对其进行简化。首先对行列式(8-4.11)施行下列运算：

第1、2行除以$-\rho_1$，$4i+1$和$4i+2$行除以$\rho_i(i=1, 2,$

$\cdots, n-1)$；

$4i-1$ 行乘以 $-(\gamma_i-1)$ 与 $4i+1$ 行 相 加 作 为 后 者新的结果；

$4i$ 行乘以 $\gamma_i$ 与 $4i+2$ 行相加作为后者新的结果；

$4i+1$ 行乘以 $-1$ 与 $4i-1$ 相加作为后者新的结果；

$4i+2$ 行乘以 $-1$ 与 $4i$ 行相加作为后者新的结果；

用 $(-1)^{n-1}\rho_1^2 C^2/\gamma_n r_{\alpha n} r_{\beta n}\rho_n^2 \alpha_n^2$ 去乘最后一列。结果行列式的子矩阵就成为下面形式：

$$\Lambda_{con}^{(0)}=\begin{pmatrix} \gamma_1-1 & 0 & \gamma_1 & 0 \\ 0 & -\gamma_1 & 0 & \gamma_1-1 \end{pmatrix}. \tag{8-4.12}$$

$$\Lambda_{oce}^{(0)}=\begin{pmatrix} \gamma_1-1 & -j\rho_0 \mathrm{tg}P_0/P_1 r_{\alpha 0} & \gamma_1 & j\rho_0 \mathrm{tg}P_0/\rho_1 r_{\alpha 0} \\ 0 & -\gamma_1 & 0 & \gamma_1-1 \end{pmatrix}, \tag{8-4.13}$$

$$\Lambda^{(m)}=\begin{pmatrix} \cos P_m & -j\sin P_m/r_{\alpha m} & 0 & 0 & -\varepsilon_3^{(m)} & 0 & -\varepsilon_1^{(m)} & 0 \\ -jr_{\alpha m}\sin P_m & \cos P_m & 0 & 0 & 0 & \varepsilon_2^{(m)} & 0 & -\varepsilon_4^{(m)} \\ 0 & 0 & \cos Q_m & -jr_{\beta m}\sin Q_m & \varepsilon_4^{(m)} & 0 & \varepsilon_2^{(m)} & 0 \\ 0 & 0 & j\sin Q_m/r_{\beta m} & -\cos Q_m & 0 & -\varepsilon_1^{(m)} & 0 & \varepsilon_3^{(m)} \end{pmatrix}, \tag{8-4.14}$$

$$
A^{(n-1)}=\begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & \varepsilon_3^{(n-1)} & -\varepsilon_1^{(n-1)} r_{\beta n}\varepsilon \\
\cdot & \cdot & \cdot & \cdot & \varepsilon_2^{(n-1)} r_{\alpha n} & -\varepsilon_4^{(n-1)}\varepsilon \\
\cdot & \cdot & \cdot & \cdot & \varepsilon_4^{(n-1)} & \varepsilon_2^{(n-1)} r_{\beta n}\varepsilon \\
\cdot & \cdot & \cdot & \cdot & -\varepsilon_1^{(n-1)} r_{\beta n} & \varepsilon_3^{(n-1)}\varepsilon
\end{pmatrix}.
$$

(8-4.15)

$A^{(n-1)}$前四列与$A^{(m)}$对应的元素形式相同，只须将下角标$m$改为$n-1$。式中有关参数$\varepsilon_1^{(n)}$，$\varepsilon_2^{(n)}$，$\varepsilon_3^{(n)}$，$\varepsilon_4^{(n)}$由表8-4-1给出，且

$$
\varepsilon=(-1)^{n-1}\rho_1^2 c^2/\gamma_{\alpha n} r_{\beta n}\rho_n^2 \alpha_n^2 .
$$

用Laplace分解定理将频散函数分解。最后可将频散函数(8-4.11)分解为一系列子矩阵的乘积形式

$$
F_R(\omega,\ c)=T^{(0)}\overline{F}^{(1)}F^{(2)}\overline{F}^{(3)}\cdots
\begin{cases}
F^{(n-2)}\overline{F}^{(n-1)}T^{(n)} & \text{当 } n \text{ 为偶数,}\\
\overline{F}^{(n-2)}F^{(n-1)}\overline{T}^{(n)} & \text{当 } n \text{ 为奇数.}
\end{cases}
$$

(8-4.16)

(8-4.16)的符号矩阵形式是$(1\times 6)(6\times 6)\cdots\cdots(6\times 6)(6\times 1)$，各个子矩阵的元素是

$$
T^{(0)}=[-\gamma_1(\gamma_1-1),0,(\gamma_1-1)^2,\gamma_1^2,q/C^2\rho_1,\gamma_1(\gamma_1-1)].
$$

(8-4.17)

上式中，对于大陆模型$q=0$，对于海洋模型$q=j\rho_0 c^2 \mathrm{tg}P_0/r_{\alpha 0}$。

$$
F^{(m)}=\begin{pmatrix}
F_{1212}^{(m)} & F_{1213}^{(m)} & F_{1214}^{(m)} & F_{1223}^{(m)} & F_{1224}^{(m)} & F_{1234}^{(m)} \\
F_{1312} & F_{1313} & F_{1314} & F_{1323} & F_{1324} & F_{1334} \\
F_{1412} & F_{1413} & F_{1414} & F_{1423} & F_{1424} & F_{1434} \\
F_{2312} & F_{2313} & F_{2314} & F_{2323} & F_{2324} & F_{2334} \\
F_{2412} & F_{2413} & F_{2414} & F_{2423} & F_{2424} & F_{2434} \\
F_{3412} & F_{3413} & F_{3414} & F_{3423} & F_{3424} & F_{3434}
\end{pmatrix}. \tag{8-4.18}
$$

$$
F^{(m)}=\begin{pmatrix}
F_{3434}^{(m)} & -F_{3424}^{(m)} & F_{3423}^{(m)} & F_{3414}^{(m)} & -F_{3413}^{(m)} & F_{3412}^{(m)} \\
-F_{2434} & F_{2424} & -F_{2423} & -F_{2414} & F_{2413} & -F_{2412} \\
F_{2334} & -F_{2324} & F_{2323} & F_{2314} & -F_{2313} & F_{2312} \\
F_{1434} & -F_{1424} & F_{1423} & F_{1414} & -F_{1413} & F_{1412} \\
-F_{1334} & F_{1324} & -F_{1323} & -F_{1314} & F_{1313} & -F_{1312} \\
F_{1234} & -F_{1224} & F_{1223} & F_{1214} & -F_{1213} & F_{1212}
\end{pmatrix} \tag{8-4.19}
$$

$$
T^{(n)}=\begin{pmatrix} 0 \\ -r_{\alpha n} \\ r_{\alpha n}r_{\beta n} \\ 1 \\ -r_{\beta n} \\ 0 \end{pmatrix}, \qquad
\overline{T}^{(n)}=\begin{pmatrix} 0 \\ r_{\beta n} \\ 1 \\ r_{\alpha n}r_{\beta n} \\ r_{\alpha n} \\ 0 \end{pmatrix}. \tag{8-4.20}
$$

以上各元素 $F_{ijkl}^{(m)}$ 可参见表8-4-1。

在频散函数的矩阵分解式(8-4.16)中，（1 × 6）矩阵的第一个元素总是第六个元素的负值。（6 × 6）界面子矩阵的对称性可表示如下

$[U^{(m+1)},\ jV^{(m+1)},\ W^{(m+1)},\ R^{(m+1)},\ jS^{(m+1)},\ -U^{(m+1)}]$

表8-4-1　　元素 $F^{(m)}_{ijkl}$的表达式

| ij \ kl | 12 | 13 | 14 | 23 | 24 | 34 |
|---|---|---|---|---|---|---|
| 12 | $-\varepsilon_8^{(m)}$ | 0 | $\varepsilon_{13}^{(m)}$ | $\varepsilon_6^{(m)}$ | 0 | $\varepsilon_{10}^{(m)}$ |
| 13 | $-j(\varepsilon_{11}\zeta_9+\varepsilon_7\zeta_{10})$ | $\varepsilon_{15}\zeta_{14}$ | $j(\varepsilon_{14}\zeta_9+\varepsilon_{12}\zeta_{10})$ | $j(\varepsilon_9\zeta_9+\varepsilon_5\zeta_{10})$ | $-\varepsilon_{15}\zeta_7$ | $j(\varepsilon_{11}\zeta_9+\varepsilon_7\zeta_{10})$ |
| 14 | $\varepsilon_{11}\zeta_7-\varepsilon_7\zeta_{12}$ | $j(\varepsilon_{15}\zeta_{10})$ | $-\varepsilon_{14}\zeta_7+\varepsilon_{12}\zeta_{12}$ | $-\varepsilon_9\zeta_7+\varepsilon_5\zeta_{12}$ | $j(\varepsilon_{15}\zeta_8)$ | $-\varepsilon_{11}\zeta_7+\varepsilon_7\zeta_{12}$ |
| 23 | $-\varepsilon_{11}\zeta_{15}+\zeta_7\varepsilon_7$ | $j(\varepsilon_{15}\zeta_9)$ | $\varepsilon_{14}\zeta_{15}-\varepsilon_{12}\zeta_7$ | $\varepsilon_9\zeta_{15}-\varepsilon_5\zeta_7$ | $j(\varepsilon_{15}\zeta_{11})$ | $\varepsilon_{11}\zeta_{15}-\varepsilon_7\zeta_7$ |
| 24 | $-j(\varepsilon_{11}\zeta_{11}+\varepsilon_7\zeta_8)$ | $-\varepsilon_{15}\zeta_7$ | $j(\varepsilon_{14}\zeta_{11}+\varepsilon_{12}\zeta_8)$ | $j(\varepsilon_9\zeta_{11}+\varepsilon_5\zeta_8)$ | $\varepsilon_{15}\zeta_{13}$ | $j(\varepsilon_{11}\zeta_{11}+\varepsilon_7\zeta_8)$ |
| 34 | $\varepsilon_{10}$ | 0 | $-\varepsilon_{13}$ | $-\varepsilon_6$ | 0 | $-\varepsilon_8$ |

$\varepsilon_0^{(m)}=\rho_{m+1}/\rho_m$

$\varepsilon_1=\gamma_m-\varepsilon_0\gamma_{m+1}$

$\varepsilon_2=\varepsilon_1-1$

$\varepsilon_3=\varepsilon_1+\varepsilon_0$

$\varepsilon_4=\varepsilon_2+\varepsilon_0$

$\varepsilon_5=\varepsilon_1^2$

$\varepsilon_6=\varepsilon_1\varepsilon_2$

$\varepsilon_7=\varepsilon_1\varepsilon_3$

$\varepsilon_8^{(m)}=\varepsilon_1^{(m)}\varepsilon_4^{(m)}$

$\varepsilon_9=\varepsilon_2{}^2$

$\varepsilon_{10}=\varepsilon_2\varepsilon_3$

$\varepsilon_{11}=\varepsilon_2\varepsilon_4$

$\varepsilon_{12}=\varepsilon_3^2$

$\varepsilon_{13}=\varepsilon_3\varepsilon_4$

$\varepsilon_{14}=q^2{}_4$

$\varepsilon_{15}=-\varepsilon_0$

$\varepsilon_{16}=\varepsilon_8+\varepsilon_{10}$

$\zeta_1^{(m)}=\cos P_m$

$\zeta_2=\cos Q_m$

$\zeta_3=r\alpha_m\sin P_m$

$\zeta_4=\sin P_m/r_{\alpha m}$

$\zeta_5=r\beta_m\sin Q_m$

$\zeta_6=\sin Q_m/r_{\beta m}$

$\zeta_7=\zeta_1\zeta_2$

$\zeta_8=\zeta_1\zeta_5$

$\zeta_9^{(m)}=\zeta_1^{(m)}\zeta_6^{(m)}$

$\zeta_{10}=\zeta_2\zeta_3$

$\zeta_{11}=\zeta_2\zeta_4$

$\zeta_{12}=\zeta_3\zeta_5$

$\zeta_{13}=\zeta_4\zeta_5$

$\zeta_{14}=\zeta_3\zeta_6$

$\zeta_{15}=\zeta_4\zeta_6$

$$
=[U^{(m)},\ jV^{(m)},\ W^{(m)},\ R^{(m)},\ jS^{(m)},\ -U^{(m)}]
\begin{pmatrix}
\delta & 0 & \nu & \upsilon & 0 & \eta \\
jk & \cdot & \cdot & \cdot & \cdot & -jk \\
\theta & \cdot & \cdot & \cdot & \cdot & -\theta \\
\phi & \cdot & \cdot & \cdot & \cdot & -\phi \\
jl & \cdot & \cdot & \cdot & \cdot & -jl \\
\eta & 0 & -\nu & -\upsilon & 0 & \delta
\end{pmatrix}. \tag{8-4.21}
$$

上式中，（6 × 6）矩阵中第2及第5列的第1和第6个元素为零，这意味着在求（1 × 6）矩阵相应的元素时，只作4次而不是6次乘法。（1 × 6）乘积矩阵的其余四个元素只作5次乘法，这是由于第一个元素和第六个元素等值反号。因此在求(8-4.21)矩阵乘法时，只要作两个4次乘法和三个5次乘法，这是对原来六个6次乘法的重要改进。如果把这五个元素用解析式写出来，就会见到它还可以用简单的代数因式分解进行简化。Kopoff 快速计算频散函数的关键部分是不断由层参数去递推新的矩阵元素，即根据 $m$ 层的矩阵元素推出 $m+1$ 层的矩阵元素。当 $m+1$ 为偶数时有

$$
U^{(m+1)}=-\varepsilon_{16}^{(m+1)}U^{(m)}+\varepsilon_{11}^{(m+1)}K^{(m+1)}+\varepsilon_{7}^{(m+1)}L^{(m+1)},
$$

$$
V^{(m+1)}=\varepsilon_{15}^{(m+1)}(\zeta_{14}^{(m+1)}V^{(m)}+\zeta_{10}^{(m+1)}W^{(m)}+\zeta_{9}^{(m+1)}R^{(m)}
$$

$$
-\zeta_{7}^{(m+1)}S^{(m)}),
$$

$$
W^{(m+1)}=-\varepsilon_{14}^{(m+1)}K^{(m+1)}-\varepsilon_{12}^{(m+1)}L^{(m+1)}
$$

$$
+2\varepsilon_{13}^{(m+1)}U^{(m)},
$$

$$
R^{(m+1)}=-\varepsilon_{9}^{(m+1)}K^{(m+1)}-\varepsilon_{5}^{(m+1)}L^{(m+1)}
$$

$$
+2\varepsilon_{8}^{(m+1)}U^{(m)},
$$

$$S^{(m+1)}=\varepsilon_{15}^{(m+1)}\left(-\zeta_{7}^{(m+1)}V^{(m)}+\zeta_{8}^{(m+1)}W^{(m)}\right.$$

$$\left.+\zeta_{11}^{(m+1)}R^{(m)}+\zeta_{13}^{(m+1)}S^{(m)}\right). \qquad (8\text{-}4.22)$$

其中

$$K^{(m+1)}=\zeta_{9}^{(m+1)}V^{(m)}+\zeta_{7}^{(m+1)}W^{(m)}-\zeta_{15}^{(m+1)}R^{(m)}$$

$$+\zeta_{11}^{(m+1)}S^{(m)},$$

$$L^{(m+1)}=\zeta_{10}^{(m+1)}V^{(m)}-\zeta_{12}^{(m+1)}W^{(m)}+\zeta_{7}^{(m+1)}R^{(m)}$$

$$+\zeta_{8}^{(m+1)}S^{(m)}.$$

当$m+1$为奇数时有

$$U^{(m+1)}=-\varepsilon_{16}^{(m+1)}U^{(m)}+\varepsilon_{11}^{(m+1)}X^{(m+1)}+\varepsilon_{7}^{(m+1)}Z^{(m+1)},$$

$$V^{(m+1)}=\varepsilon_{15}^{(m+1)}\left(\zeta_{13}^{(m+1)}V^{(m)}-\zeta_{11}^{(m+1)}W^{(m)}\right.$$

$$\left.-\zeta_{8}^{(m+1)}R^{(m)}-\zeta_{7}^{(m+1)}S^{(m)}\right),$$

$$W^{(m+1)}=\varepsilon_{9}^{(m+1)}X^{(m+1)}+\varepsilon_{5}^{(m+1)}Z^{(m+1)}-2\varepsilon_{6}^{(m+1)}U^{(m)},$$

$$R^{(m+1)}=\varepsilon_{14}^{(m+1)}X^{(m+1)}+\varepsilon_{12}^{(m+1)}Z^{(m+1)}-2\varepsilon_{13}^{(m+1)}U^{(m)},$$

$$S^{(m+1)}=\varepsilon_{15}^{(m+1)}\left(-\zeta_{7}^{(m+1)}V^{(m)}-\zeta_{9}^{(m+1)}W^{(m)}\right.$$

$$\left.-\zeta_{10}^{(m+1)}R^{(m)}+\zeta_{14}^{(m+1)}S^{(m)}\right). \qquad (8\text{-}4.23)$$

其中

$$X^{(m+1)}=\zeta_{11}^{(m+1)}V^{(m)}+\zeta_{15}^{(m+1)}W^{(m)}-\zeta_{7}^{(m+1)}R^{(m)}$$

$$+\zeta_{9}^{(m+1)}S^{(m)},$$

$$Z^{(m+1)}=\zeta_{8}^{(m+1)}V^{(m)}-\zeta_{7}^{(m+1)}W^{(m)}+\zeta_{12}^{(m+1)}R^{(m)}$$

$$+\zeta_{10}^{(m+1)}S^{(m)}.$$

频散函数是由以下实数初值递推的：

$$
\begin{aligned}
&U^{(0)}=-\gamma_1(\gamma_1-1),\\
&V^{(0)}=0,\\
&W^{(0)}=(\gamma_1-1)^2, \qquad\qquad (8\text{-}4.24)\\
&R^{(0)}=\gamma_1^2,\\
&S^{(0)}=\begin{cases}0 & \text{对于大陆模型，}\\ \rho_0\,\mathrm{tg}P_0/\rho_1 r_{a0} & \text{对于海洋模型。}\end{cases}
\end{aligned}
$$

将(8-4.24)代入(8-4.22)及(8-4.23)反复递推，直到频散函数计算到 $(n-1)$ 界面为止。

$$
\begin{aligned}
&[U^{(n-1)},\ jV^{(n-1)},\ W^{(n-1)},\ R^{(n-1)},\ jS^{(n-1)},\\
&\quad -U^{(n-1)}]=T^{(0)}\overline{F}^{(1)}F^{(2)}\cdots\cdots\\
&\quad\begin{cases}F^{(n-1)} & \text{当 } n-1 \text{ 为偶数，}\\ \overline{F}^{(n-1)} & \text{当 } n-1 \text{ 为奇数。}\end{cases} \qquad\qquad (8\text{-}4.25)
\end{aligned}
$$

最后的频散函数为

$$
F_R(\omega,\ c)=[V^{(n-1)},\ W^{(n-1)},\ R^{(n-1)},\ S^{(n-1)}]
\begin{cases}
\begin{pmatrix}-(1-C^2/\alpha_n^2)^{1/2}\\ -(1-C^2/\alpha_n^2)^{1/2}(1-C^2/\beta_n^2)^{1/2}\\ 1\\ -(1-C^2/\beta_n^2)^{1/2}\end{pmatrix}\varepsilon & \text{若 } n \text{ 为偶数，}\\[2ex]
\begin{pmatrix}(1-C^2/\beta_n^2)^{1/2}\\ 1\\ -(1-C^2/\alpha_n^2)^{1/2}(1-C^2/\beta_n^2)^{1/2}\\ (1-C^2/\alpha_n^2)^{1/2}\end{pmatrix}\varepsilon & \text{若 } n \text{ 为奇数。}
\end{cases}
$$

(8-4.26)

由于(8-4.22)—(8-4.24)及(8-4.26)仅包含实数，故求频散函数时完全避免了复数运算，易于在计算机上实现。

## （二）最深界面在外核中的固体-液体模型

这时层状结构的下半空间为均匀液体，其频散函数具有如下形式：

$$F_R(\omega,\ c)=T^{(0)}\overline{F}^{(1)}F^{(2)}$$

$$\cdots\begin{cases}\overline{F}^{(m-1)}G^{(m)}\overline{H}^{(m-1)}H^{(m-2)} \\ F^{(m-1)}\overline{G}^{(m)}H^{(m-1)}\overline{H}^{(m-2)}\end{cases}\cdots \quad \begin{matrix} m\text{为偶数,} \\ m\text{为奇数.}\end{matrix}$$

$$\cdots\begin{cases}H^{(n-2)}\overline{H}^{(n-1)}T_{liq}^{(n)} & n\text{为偶数,} \\ \overline{H}^{(n-2)}H^{(n-1)}\overline{T}_{liq}^{(n)} & n\text{为奇数.}\end{cases} \tag{8-4.27}$$

式中 $T^{(0)}$，$\overline{F}^{(1)}\cdots$，$\overline{F}^{(m-1)}$ 的意义同前，$liq$ 表示液体（*liquid*），矩阵 $G^{(m)}$ 或者 $\overline{G}^{(m)}$ 表示幔核边界（第 $m$ 界面），由下式给出：

$$G^{(m)}=\begin{pmatrix} G_{121}^{(m)} & G_{122}^{(m)} \\ G_{131} & G_{132} \\ G_{141} & G_{142} \\ G_{231} & G_{232} \\ G_{241} & G_{242} \\ G_{341} & G_{342}\end{pmatrix},\quad \overline{G}^{(m)}=\begin{pmatrix} -jG_{342}^{(m)} & jG_{341}^{(m)} \\ jG_{242} & -jG_{241} \\ -jG_{232} & jG_{231} \\ -jG_{142} & jG_{141} \\ jG_{132} & -jG_{131} \\ -jG_{122} & jG_{121}\end{pmatrix}. \tag{8-4.28}$$

式中各元素为

$$G_{121}^{(m)}=0,\qquad G_{122}^{(m)}=j\gamma_m(\gamma_m-1),$$

$$G_{131}^{(m)}=-j\varepsilon^{(m)}\zeta_{14}^{(m)},\qquad G_{132}^{(m)}=-(\gamma_m-1)^2\zeta_9^{(m)}-\gamma_m^2\zeta_{10}^{(m)},$$

$$G_{141}^{(m)}=\varepsilon^{(m)}\zeta_{10}^{(m)},\qquad G_{142}^{(m)}=-j[(\gamma_m-1)^2\zeta_7^{(m)}-\gamma_m^2\zeta_{12}^{(m)}],$$

$$G_{231}^{(m)}=\varepsilon_0^{(m)}\zeta_9^{(m)}, \qquad G_{232}^{(m)}=j[(\gamma_m-1)^2\zeta_{15}^{(m)}-\gamma_m^2\zeta_7^{(m)}],$$

$$G_{241}^{(m)}=j\varepsilon_0^{(m)}\zeta_7^{(m)}, \qquad G_{242}^{(m)}=-(\gamma_m-1)^2\zeta_{11}^{(m)}-\gamma_m^2\zeta_8^{(m)},$$

$$G_{341}^{(m)}=0, \qquad G_{342}^{(m)}=-j\gamma_m(\gamma_m-1).$$

(8-4.29)

对于液体界面情况，可由下列$2\times2$阶子矩阵$H^{(m)}$和$\overline{H}^{(m)}$表示：

$$H^{(m)}=\begin{pmatrix} H_{11}^{(m)} & H_{12}^{(m)} \\ H_{21}^{(m)} & H_{22}^{(m)} \end{pmatrix}, \quad \overline{H}^{(m)}=\begin{pmatrix} -H_{22}^{(m)} & H_{21}^{(m)} \\ H_{12}^{(m)} & -H_{11}^{(m)} \end{pmatrix}。$$

(8-4.30)

其中

$$H_{11}^{(m)}=\varepsilon_0^{(m)}\zeta_3^{(m)}, \qquad H_{12}^{(m)}=-j\zeta_1^{(m)},$$

$$H_{21}^{(m)}=j\varepsilon_0^{(m)}\zeta_1^{(m)}, \qquad H_{22}^{(m)}=-\zeta_4^{(m)}.$$

(8-4.31)

最下层界面

$$T_{1iq}^{(n)}=\begin{bmatrix} -jr_{an} \\ j \end{bmatrix}, \qquad \overline{T}_{1iq}^{(n)}=\begin{bmatrix} 1 \\ r_{an} \end{bmatrix}.$$

(8-4.32)

这种模型的计算过程是，首先计算前$m-1$个界面矩阵的元素。

$$[U^{(m-1)}, jV^{(m-1)}, w^{(m-1)}, R^{(m-1)}, jS^{(m-1)}, -U^{(m-1)}]=T^{(0)}\overline{F}^{(0)}F^{(2)}\cdots$$

$$\begin{cases} F^{(m-1)} & m-1\text{ 为偶数}, \\ F^{(m-1)} & m-1\text{ 为奇数}. \end{cases}$$

对于第 $m$ 个界面，其乘积矩阵具有下列形式：

$$[\Psi^{(m)},\ j\Phi^{(m)}]=[U^{(m-1)},\ jV^{(m-1)},\ W^{(m-1)},$$

$$R^{(m-1)},\ jS^{(m-1)},\ -U^{(m-1)}]\begin{cases}\begin{pmatrix} o & j\Phi \\ jv & l \\ \nu & j\eta \\ k & j\delta \\ j\theta & \xi \\ o & -j\phi \end{pmatrix}^{(m)} & m\text{ 为偶数}, \\ \begin{pmatrix} -\phi & o \\ j\xi & \theta \\ \delta & jk \\ \eta & j\nu \\ jl & v \\ \phi & o \end{pmatrix}^{(m)} & m\text{ 为奇数}. \end{cases}$$

(8-4.33)

这样,其积可以仅用实数表示，若 $m$ 为偶数，

$$\Psi^{(m)}=-vV^{(m-1)}+\ vW^{(m-1)}+kR^{(m-1)}-\theta S^{(m-1)},$$

$$\Phi^{(m)}=2\phi U^{(m-1)}+lV^{(m-1)}+\eta W^{(m-1)}+\xi S^{(m-1)}.$$

(8-4.34)

当 $m$ 为奇数时有

$$\Psi^{(m)}=-2\phi U^{(m-1)}-\xi V^{(m-1)}+\delta W^{(m-1)}+\eta R^{(m-1)}$$
$$-jS^{(m-1)},$$
$$\Phi^{(m)}=\theta V^{(m-1)}+kW^{(m-1)}+\nu R^{(m-1)}+vS^{(m-1)}.$$

(8-4.35)

构造好矩阵 $[\Psi^{(m)},\ j\Phi^{(m)}]$ 后，频散函数就可以表示成下面的符号矩阵形式：

$$F_R(\omega,c)=[R,I]^{(m)}\begin{bmatrix} R & I \\ I & R \end{bmatrix}^{(m+1)}\cdots\begin{bmatrix} R & I \\ I & R \end{bmatrix}^{(n-1)}\begin{bmatrix} R \\ I \end{bmatrix}^{(n)}. \tag{8-4.36}$$

利用后面式(8-5.12)—(8-5.14)所给出的方法，可以避免复数的运算。

### （三） 最深界面在内核中的固体-液体-固体模型

这时频散函数取如下形式：

$$F_R(\omega,\ c)=T^{(0)}F^{(1)}F^{(2)}\cdots\begin{Bmatrix} \overline{F}^{(m-1)}G^{(m)}\overline{H}^{(m+1)} \\ F^{(m-1)}\overline{G}^{(m)}H^{(m+1)} \end{Bmatrix}\cdots$$

$$\begin{matrix} m\text{为偶数} \\ m\text{为奇数} \end{matrix}\cdots\begin{Bmatrix} \overline{H}^{(p-1)}I^{(p)}\overline{F}^{(p+1)}F^{(p+2)} \\ H^{(p-1)}\overline{I}^{(p)}F^{(p+1)}\overline{F}^{(p+2)} \end{Bmatrix}\cdots\begin{matrix} p\text{为偶数} \\ p\text{为奇数} \end{matrix}$$

$$\cdots\begin{cases} \overline{F}^{(n-1)}T_{sol}^{(n)} & n\text{为偶数}, \\ F^{(n-1)}\overline{T}_{sol}^{(n)} & n\text{为奇数}. \end{cases} \tag{8-4.37}$$

矩阵 $I^{(p)}$ 或 $\overline{I}^{(p)}$ 表示为液体-固体接触的边界，$sol$ 表示固体($solid$)，此界面子矩阵为

$$I^{(m)}=\begin{pmatrix} I_{112}^{(m)} & I_{113}^{(m)} & I_{114}^{(m)} & I_{123}^{(m)} & I_{124}^{(m)} & I_{134}^{(m)} \\ I_{212}^{(m)} & I_{213}^{(m)} & I_{214}^{(m)} & I_{223}^{(m)} & I_{224}^{(m)} & I_{234}^{(m)} \end{pmatrix},$$

$$\overline{I}^{(m)}=\begin{pmatrix} jI_{234}^{(m)} & -jI_{224}^{(m)} & jI_{223}^{(m)} & jI_{214}^{(m)} & -jI_{213}^{(m)} & jI_{212}^{(m)} \\ -jI_{134}^{(m)} & jI_{124}^{(m)} & -jI_{123}^{(m)} & -jI_{114}^{(m)} & jI_{113}^{(m)} & -jI_{112}^{(m)} \end{pmatrix}. \tag{8-4.38}$$

其中

$$I_{112}^{(m)}=-\varepsilon_0^{(m)}\gamma_{m+1}(\gamma_{m+1}-1)\xi_3^{(m)},\ I_{212}^{(m)}$$

$$=-j\varepsilon_0^{(m)}\gamma_{m+1}(\gamma_{m+1}-1)\zeta_1^{(m)},$$

$I_{113}^{(m)}=0,$ $\qquad$ $I_{213}^{(m)}=0,$

$I_{114}^{(m)}=\varepsilon_0^{(m)}(\gamma_{m-1}-1)^2\zeta_3^{(m)},$ $\qquad$ $I_{214}^{(m)}=j\varepsilon_0^{(m)}(\gamma_{m-1}-1)^2\zeta_1^{(m)},$

$I_{123}^{(m)}=\varepsilon_0^{(m)}\gamma_{m+1}^2\zeta_3^{(m)},$ $\qquad$ $I_{223}^{(m)}=j\varepsilon_0^{(m)}\gamma_{m+1}^2\zeta_1^{(m)},$

$I_{124}^{(m)}=-j\zeta_1^{(m)},$ $\qquad$ $I_{224}^{(m)}=-\zeta_4^{(m)},$

$I_{134}^{(m)}=\varepsilon_0^{(m)}\gamma_{m+1}(\gamma_{m+1}-1)\zeta_3^{(m)},$ $\qquad$ $I_{234}^{(m)}=j\varepsilon_0^{(m)}\gamma_{m+1}(\gamma_{m+1}-1)\zeta_1^{(m)}.$

计算时，仍然按前述方式先算前 $p-1$ 个界面子矩阵乘积

$$[\Psi^{(p-1)},\ j\Phi^{(p-1)}]=T^{(0)}\overline{F}^{(1)}\cdots\begin{cases}\overline{H}^{(p-2)}H^{(p-1)} & p-1\text{为偶数},\\ H^{(p-2)}\overline{H}^{(p-1)} & p-1\text{为奇数},\end{cases}\tag{8-4.40}$$

第 $p$ 个界面矩阵乘法具有下面的形式

$$[U^{(p)},\ jV^{(p)},\ W^{(p)},\ R^{(p)},\ jS^{(p)},\ -U^{(p)}]=[\Psi^{(p-1)},\ j\Phi^{(p-1)}]\begin{pmatrix}v, & \cdots, & -v\\ j\nu, & \cdots, & -j\nu\end{pmatrix}.\tag{8-4.41}$$

在( 1 × 6 )矩阵中第一个元素与最后一个元素绝对值相等符号相反。整个运算过程避免了复数运算。

## § 8-5 层状介质中Love波的频散计算

上两节都是讨论多层介质中 Rayleigh波 频散计算问题的，现在转向 Love波 的计算。Love 波 的 情况较简单，因为每个界

面只有切向位移与切面应力连续两个边界条件。

Love 波质点运动的特征为垂直于由波的传播方向及垂直线组成的平面。层状结构坐标系统与图8-3-1相同。因为Love波不能在理想液体中传播，故图8-3-1适用于大陆和海洋两种情况。对于海洋地区，液体层可以忽略不计。这个模型实际上表示有 $n-1$ 个均匀固体层覆盖在均匀半空间之上。如果最深的界面在地幔中，则半空间是固体的，如果在核幔边界，则下半空间为液体的。

Love 波的频散函数可以由 Haskell 所导出的公式表示。下面进行简单的推导。

Love 波是 $SH$ 型横波，设其位函数为 $V$，则满足下列波动方程：

$$\nabla^2 V = \frac{1}{\beta^2}\frac{\partial^2 V}{\partial t^2}. \tag{8-5.1}$$

仿照 Rayleigh 波的推导过程，并采用 § 8-3 中的有关符号，可得第 $m$ 个界面的位移及应力分量递推公式为

$$\begin{aligned}\frac{\dot{V}_m}{C} &= \frac{\dot{V}_{m-1}}{C}\cos Q_m + j(p_{pz})_m\mu_m^{-1}\,\gamma_{\beta m}^{-1}\sin Q_m,\\ (p_{yz})_m &= j\frac{\dot{V}_{m-1}}{c}\mu_m\gamma_{\beta m}\sin Q_m + (p_{yz})_{m-1}\cos Q_m.\end{aligned} \tag{8-5.2}$$

由此可以得到界面子矩阵

$$\boldsymbol{a}_m = \begin{bmatrix}\cos Q_m & j\mu_m^{-1}\,\gamma_{\beta m}^{-1}\sin Q_m\\ j\mu_m r_{\beta m}\sin Q_m & \cos Q_m\end{bmatrix}. \tag{8-5.3}$$

按照 Rayleigh 波同样定义

$$\boldsymbol{A} = \boldsymbol{a}_{n-1}\boldsymbol{a}_{n-2}\cdots\boldsymbol{a}_1. \tag{8-5.4}$$

于是

$$\begin{pmatrix} \left(\frac{\dot{V}}{c}\right)_{n-1} \\ (p_{yz})_{n-1} \end{pmatrix} = A \begin{pmatrix} \left(\frac{\dot{V}}{c}\right)_0 \\ (p_{yz})_0 \end{pmatrix} = \begin{pmatrix} A_{11}\left(\frac{\dot{V}}{c}\right)_0 + A_{12}(p_{yz})_0 \\ A_{21}\left(\frac{\dot{V}}{c}\right)_0 + A_{22}(p_{yz})_0 \end{pmatrix}. \tag{8-5.5}$$

根据面波存在条件，设最深界面为固体-固体分界，则频散方程为

$$0 = A_{21} + \mu_n r_{\beta n} A_{11}. \tag{8-5.6}$$

其中 $n$ 表示固体半空间。如果最深界面相应为幔核边界，则频散方程为

$$0 = A_{21}. \tag{8-5.7}$$

频散曲线任一点的坐标（$\omega$, $c$）上，其函数 $F_1(\omega, c)$ 总是为零。故 Love 波频散函数可根据上二式定义为

$$F_L(\omega, c) = \begin{cases} (A_{21} + \mu_n r_{\beta n} A_{11})/i & \text{最深界面在地幔中,} \\ A_{21}/i & \text{最深界面在幔核边界.} \end{cases} \tag{8-5.8}$$

这还可以写成更简洁的矩阵方程

$$F_L(\omega, c) = [s, \ -j] \begin{bmatrix} A_{11} \\ A_{21} \end{bmatrix}. \tag{8-5.9}$$

式中 $s$ 取决于地层模型，若最深界面为固体-液体分界面，则 $s=0$；若最深界面为固体-固体分界面，则 $s = -\mu_n(1 - c^2/\beta^2{}_n)^{1/2}$。

为了提高计算效率，我们并不由（8-5.4）式计算 $A_{11}$，$A_{21}$ 的值，因为这将要做（2×2）（2×2）的矩阵运算。我们把它改为（1×2）（2×2）矩阵运算，就可以提高速度。同为频散函数正好具有这种形式：

$$F_L(\omega, c) = [s, \ -j] a_{n-1} a_{n-2} \cdots a_2 \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix}. \tag{8-5.10}$$

矩阵运算从左向右进行，每次都作(1 × 2)(2 × 2)运算。为了得到频散函数的最终形式，还需要消除复数运算。因为(8-5.10)每一个矩阵乘积都具有如下符号形式：

$$[R\ I]\begin{bmatrix} R & I \\ I & R \end{bmatrix}. \tag{8-5.11}$$

为消除复数运算可以重新定义频散函数

$$F_L(\omega, c) = [s, -1]b_{n-1}b_{n-2}\cdots b_2\begin{bmatrix} b_{11} \\ b_{21} \end{bmatrix}. \tag{8-5.12}$$

其中子矩阵 $b_m$ 仅包含实数，定义为

$$b_m = \begin{bmatrix} (a_m)_{11} & (a_m)_{12}/j \\ (a_m)_{21}/j & (a_m)_{22} \end{bmatrix}. \tag{8-5.13}$$

矩阵乘积的修正形式定义为

$$(b_m b_{m-1})_{ik} = \begin{cases} (b_m)_{il}(b_{m-1})_{lk}, & \text{若 } j+k \text{ 为奇数；} \\ (-1)^{j+1}(b_m)_{jl}(b_{m-1})_{lk}, & \text{若 } j+k \text{ 为偶数。} \end{cases} \tag{8-5.14}$$

下面讨论频散曲线的数值计算方法。因为Love波及Rayleigh波的频散方程都具有下列形式：

$$F(\omega, c) = 0.$$

固定 $\omega$ (或周期 $T$)值，用第六章给出的线性插值法或抛物线插值法求频散方程的根，即得到对应的相速度 $c$。在计算相速度 $c$ 值的同时，可按(8-1.14)式同时算出群速度

$$U = \frac{c}{1 + \dfrac{T}{c}\dfrac{dc}{dT}}.$$

式中微分项 $dc/dT$ 可用差分项 $\Delta c/\Delta T$ 来代替。

## §8-6 用面波资料反演地球内部结构

有关反演的理论和算法已在前面专门讨论过。利用面波(Rayleigh 波及 Love波)反演地壳及地幔内部结构问题可简单概述如下。

首先根据面波记录测定其相速度或群速度，由此得到某一地区的实测面波频散曲线。然后对地壳-地幔的分层作出假设，并给定各层的密度$\rho_i$，纵波速度$\alpha_i$，横波速度$\beta_i$，厚度$h_i$的初值。利用广义逆或最优化理论提供的算法，反复修改各层参数，使实测频散曲线与理论频散曲线得到最好的拟合。

以上是用面波频散曲线反演地球内部结构的一般原理。但在具体计算时，还要根据面波频散曲线及参数的特点作某些约定和简化。

### (一) 反演变量的限制

一般说来，用面波频散曲线反演地球内部模型要求计算的参量是很多的。对 Rayleigh 波来说，每层有四个待定参数($h,\rho,\alpha,\beta$)；对 Love 波每层有三个参数($h,\rho,\beta$)。因此，对于 $n$ 层模型分别有 $4n$ 或 $3n$ 个参数。当地下分层较多时，会引起反演计算中的某些困难。

理论研究表明，并不需要对所有的模型参量都作为独立变量进行反演。因为面波频散的相速度值对于纵波速度$\alpha$，横波速度$\beta$及密度$\rho$变化的敏度是不一样的。一般说来，Love 波相速度对密度的导数$\partial c/\partial\rho$比较小，往往是接近于零的值，而对横波速度变化较敏感，即$\partial c/\partial\beta$值较大。这可由图 8-6-1 看出。图中各括号内的数字表示求偏导数的深度范围(单位为公里)。

对 Rayleigh 波而言，虽然在某些周期范围内密度是一个重要的变量，但起主要作用的仍然是横波波速$\beta$，纵波速度$\alpha$对

Rayleigh 波相速度影响较小。如图8-6-2示。

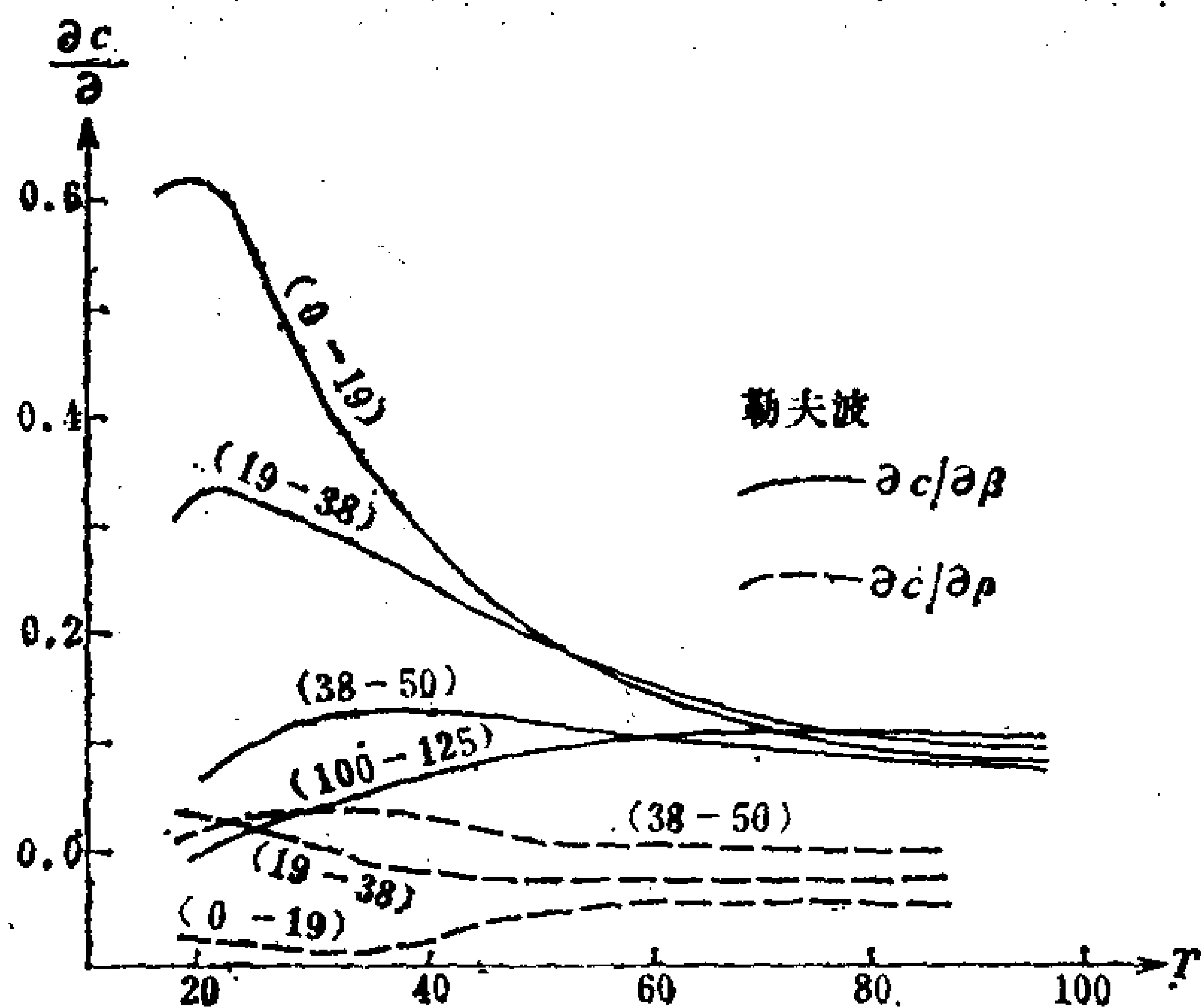



图 8-6-1

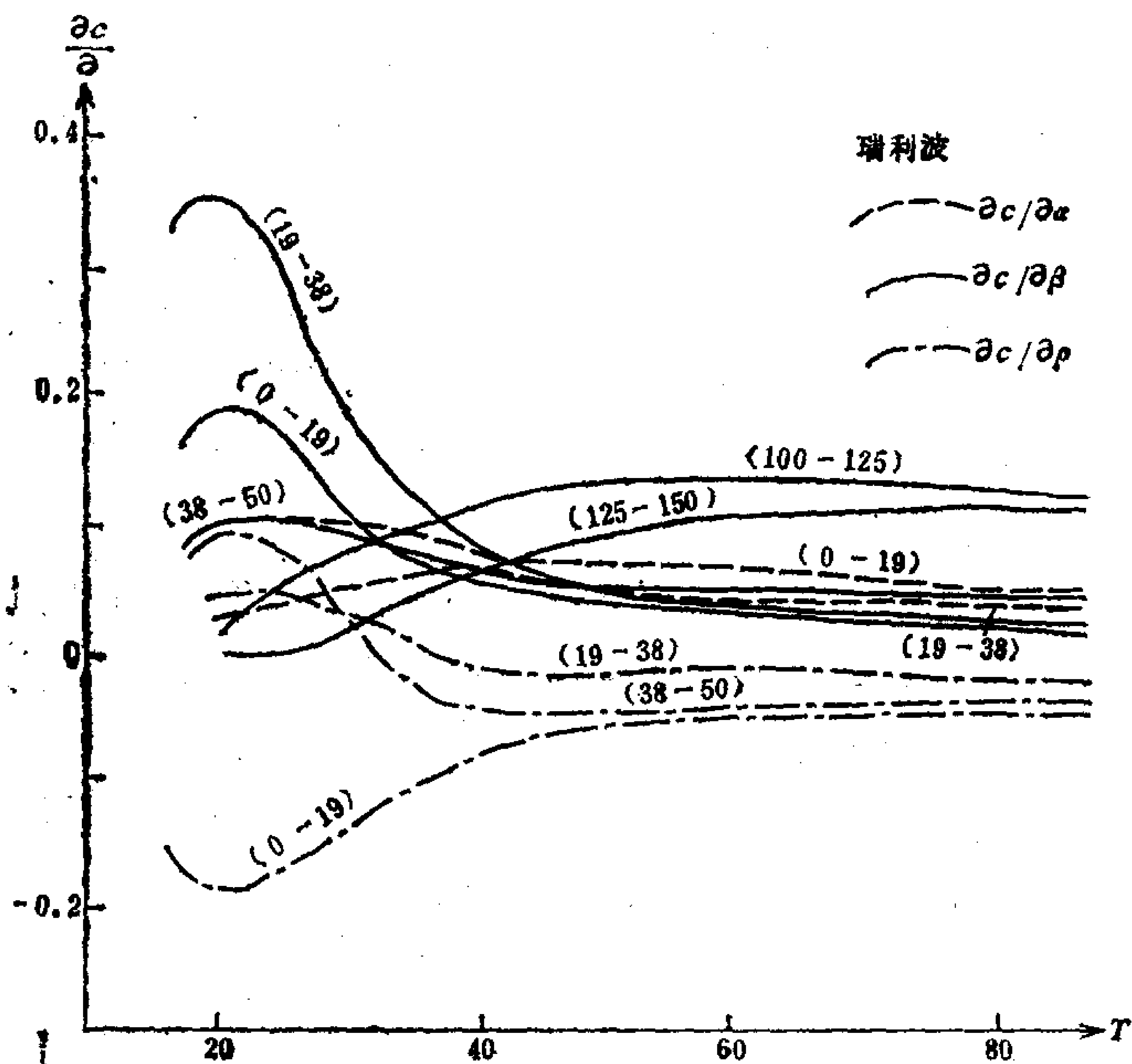



图 8-6-2

由此可见，纵波速度 $\alpha$ 与密度 $\rho$ 在面波频散曲线反演中是不太重要的。这样，每层待反演的参数实际上可以只考虑厚度 $h$ 及横波速度 $\beta$ 两个。密度 $\rho$ 及纵波速度 $\alpha$ 均作为非独立变量，根据函数约束关系由 $\beta$ 值求出。换算的办法是，设地壳及地幔顶部为 Poisson 介质(波速比取为1.73)，则各层纵波速度 $\alpha$ 均可由横波速度 $\beta$ 换算($\alpha=1.73\beta$)。在地幔低速层内，波速比可以取到1.90。

各层的密度 $\rho$ 值，可由求出的 $\alpha$ 值用插值法求得。根据大量的统计资料，密度 $\rho$ 与纵波速 $\alpha$ 的统计关系如下表所示（参看图1-7-2)：

| $\alpha$ | 3.0 | 3.5 | 4.0 | 4.5 | 5.0 | 5.5 | 6.0 | 6.5 | 7.0 | 7.5 | 8.0 | 8.5 | 9.0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\rho$ | 2.28 | 2.37 | 2.47 | 2.55 | 2.63 | 2.68 | 2.79 | 2.94 | 3.07 | 3.21 | 3.33 | 3.49 | 3.7 |

一般说来，面波的相速度(或群速度)受厚度 $h$ 的影响要比受横波速度 $\beta$ 的影响要小，即 $\partial c/\partial h$ 比 $\partial c/\partial \beta$ 要小。在反演时，常常采用两种办法。一种是将地球内部层次分细，例如分为许多2或5公里的薄层，固定层厚度 $h$，只反演各层的横波速度 $\beta$ 值。由于层次分得细，可能增多待反演的变量数目。

另一种方法是对各层厚度 $h_i$ 及横波速度 $\beta_i$ 同时进行反演。由于二个变量的量纲不一样，故必须对变量进行加权。权系数可取各变量标准误差的倒数，即

$$\Delta\beta_i' = \Delta\beta_i/\sigma_\beta,$$

$$\Delta h_i' = \Delta h_i/\sigma_h. \qquad (8\text{-}6.1)$$

或者采用相对权系数

$$\Delta\beta_i' = \Delta\beta_i,$$

$$\Delta h_i' = w\Delta h_i. \quad (8\text{-}6.2)$$

式中 $w=\sigma_\beta/\sigma_h$，当 $w=0$ 时，退化为只修定横波速度 $\beta_i$。

## （二）约束条件下的逐次归位法

根据地球物理问题实际条件对反演参数进行约束，是减小不唯一性，取得合理结果的重要办法。在对面波频散曲线进行反演中，除函数约束关系外，还采用区间约束或固定约束，对某些层参量进行限制，规定其可变范围。

在反演的迭代过程中，每一次迭代都要对约束域进行检查，即用逐次归位法使参数回到约束范围。

设地壳总厚度为

$$H_c = \sum_{i=1}^{N} h_i. \quad (8\text{-}6.3)$$

若约束条件取为

$$\hat{H}_1 \leqslant H_c \leqslant \hat{H}_2, \quad (8\text{-}6.4)$$

且 $\hat{H}_1 = H_0 - \delta,\qquad \hat{H}_2 = H_0 + \delta.$

式中 $H_0$ 为平均厚度，$\delta$ 是允许变化范围。

在反演过程中，(8-6.4) 式的约束条件受到破坏时，归位后各层的厚度取为

$$h_i^* = h_i - w_i(H_c - \hat{H}_j), \quad (8\text{-}6.5)$$

$$w_i = h_i/H_0,\qquad j=1 \text{ 或 } 2.$$

同样地，地壳横波平均速度为

$$\bar{\beta} = \sum_{i=1}^{N} h_i\beta_i \Big/ \sum_{i=1}^{N} h_i. \quad (8\text{-}6.6)$$

其约束条件为

$$\widehat{\beta}_1 \leqslant \overline{\beta} \leqslant \widehat{\beta}_2, \tag{8-6.7}$$

且 $\beta_1 = \beta_0 - \delta, \qquad \beta_2 = \beta_0 + \delta.$

式中 $\beta_0$ 是平均速度，$\delta$ 是允许变动范围。

当 (8-6.7) 式的约束条件受到破坏时，归位后的各层波速应按下式计算：

$$\beta_l^* = \beta_l - w_l(\overline{\beta} - \widehat{\beta}_j), \tag{8-6.8}$$

$$w_l = \beta_l / \overline{\beta}, \qquad j = 1, 2.$$

采用上述逐次归位法，计算简单可行，收敛较快。详细的方法原理第三章已作介绍。

## （三） 数值方法的讨论

在寻查目标函数极小值时，采用了收敛速度较快的阻尼最小二乘法。为了选择合适的阻尼因子，可根据对称矩阵 $A^TA$（$A$ 为 Jacobi 矩阵）的迹

$$\mathrm{Tr}\,(A^TA) = \sum_{i=1}^{n} a_{ii}^2 .$$

来确定阻尼因子的初值

$$\nu = \mathrm{Tr}(A^TA)/mn. \tag{8-6.9}$$

反演过程中可按下述两个标准而停止迭代。一是参数 $\chi$ 修正量的相对值小于某一 $\varepsilon_1$ 值，

$$\frac{\sum_{i=1}^{n} \left[\left|\Delta\chi_i\right|\right]}{\sum_{i=1}^{n} \chi_i} \leqslant \varepsilon_1. \tag{8-6.10}$$

另一是目标函数(残差平方和)小于某一 $\varepsilon_2$ 值，

$$\Phi(\hat{X}) = \sum_{i=1}^{m}[Y_i - F_i(\hat{X})]^2 \leqslant \varepsilon_2. \qquad (8\text{-}6.11)$$

$\varepsilon_1$ 及 $\varepsilon_2$ 可根据实测频散曲线的精度或模型参数的容许误差而事先给定。

最后分别计算协方差矩阵、信息矩阵 $S$ 及分辨矩阵 $R$。

# §8-7 Love 波和 Rayleigh 波频散曲线正、反演程序

## （一）Love波频散曲线正演程序

下面给出一个用 FORTRAN 语言编写的Love 波频散曲线正演计算程序。使用时仅需输入地下各层介质参数（厚度、横波速度及密度），即可同时算出相速度 $c$ 及群速度 $U$ 的频散曲线。

**（一）主要标识符及子程序说明**

THKNES(50)——各层介质厚度 $d_m$ 的数组，最多可表示50层。

BETA(50)——各层横波速度 $\beta_m$ 的数组。

RHO(50)——各层密度 $\rho_m$ 的数组

T(30)——计算频散曲线时所取周期的数组。

VPH(30)——计算相速度 $c$ 的数组。

VGR(30)——计算群速度 $U$ 的数组。

OMEGA——圆频率 $\omega = 2\pi/T$。

DCDT——表示 $\partial C/\partial T$，用于求群速度。

N——介质总的分层数。

PREA——求相速度 $c$ 时的初值。

ROOT(H,E,RT)——用线性插值法求频散方程的根的子程序。其中各虚元为：

H——搜索根自变量所取的步长。

E——求根的精度要求。

RT——所求的根。

FLOVE(C)——计算 Love 波频散函数 $F_L(\omega, c)$ 的子程序，其中虚元 c 为待求的相速度。这一子程守中用到的主要符号意义为：

BSQ(M) $= \beta_m^2$,

THKNES(M) $= d_m$,

RHQ(m) $= \rho_m$,

BETNSQ $= \beta_n^2$,

NMNUS2 $= N - 2$,

BSQRHN $= \beta_n^2 \rho_n = \mu_n$,

RBNMN $= -(1 - c^2/\beta_n^2)\mu_n$。

（二）FORTRAN-77语言源程序：

```
C        THIS PROGRAM IS DESIGNED TO COMPUTE
C           THE DISPERSION
C        CURVES FOR LOVE WAVES IN MULTILAYERED
         FLAT-STRUCTURE
C
         IMPLICIT DOUBLE PRECISION(A-H, O-Z)
         DIMENSION THKNES(50), BETA(50), RHO(50),
            T(30), VPH(30), VGR(30)
         COMMON/NP/N, PREA/TO/THKNES, BETA, RHO,
            OMEGA
         OPEN(1, FILE='PERIOD')
         READ(1, *)(T(I), I=1, 30)
   1000  PAUSE'INPUT N'
         READ*, N
         OPEN(2, FILE='TBR')
         READ(2, *)(THKNES(I),BETA(I), RHO(I), I=1,
            N)
```

```
      PREA = BETA(1)
      WRITE(*, 300)
  300 FORMAT(//8X, 'COMPUTATION OF THE
         THEORETIC VALUE FOR LOVE WAVE
         DISPERSION'//10X, 'THICKNESS', 10X, 'SHE
         ARWAVE VELOCITY', 10X, 'DEN 2SITY'/)
      WRITE(*, 400)(THKNES(I), BETA(I), RHO(I),
         I = 1, N)
  400 FORMAT(/11X, F7.3, 19X, F6.2, 16X, F6.2)
      WRITE(*, 500)
  500 FORMAT(15X,'PERIOD',8X,'PHASE VELOCITY'
         12X, 'GROUP VELOCITY'/)
      DO 200 I = 1, 30
      OMEGA = 6.283185307/T(I)
      CALL ROOT(0.05, 1E-8, VPH(I))
      OMEGA = 6.283185307/(T(I) - 0.01)
      CALL ROOT(0.05, 1E-8, VPHA)
      OMEGA = 6.283185307/(T(I) + 0.01)
      CALL ROOT(0.05, 1E-8, VPHB)
      DCDT = (VPHB - VPHA)/0.02
      VGR(I) = VPH(I)/(1 + T(I)/VPH(I) * DCDT)
      PREA = VPH(I) - 0.1
  200 CONTINUE
      WRITE(*, 100)(T(I), VPH(I), VGR(I), I = 1, 30)
  100 FORMAT(10X, F10.1, 9X, F12.9, 14XF12.9)
      GO  TO 1000
      END
C
C       THIS IS SUBPROGRAM FOR SEEK THE ROOTS
C          OF DISPERSION EQUATION
C
      SUBROUTINE ROOT (H, E, RT)
```

```
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/NP/N, PREA
      W=0.0
      X=PREA
      F0=FLOVE(X)
    4 X=X+H
      F1=FLOVE(X)
      IF(F1/F0.GT.0.0)GO TO 101
      FA=F0
      FB=F1
      A=X-H
      B=X
    1 C=B-(A-B)*FB/(FA-FB)
      IF(W)5, 7, 5
    7 W=1.0
      GO TO 6
    5 IF(DABS(C1-C).LT.E)GO TO 3
    6 C1=C
      FC=FLOVE(C)
      IF(FC*FA.GT.0.0)GO TO 2
      FB=FC
      B=C
      GO TO 1
    2 FA=FC
      A=C
      GO TO 1
    3 RT=C
      GO TO 102
  101 F0=F1
      IF(X-10.0)4, 102, 102
  102 RETURN
      END
```

```
C
C      THIS IS SUBPROGRAM FOR COMPUTATION OF
C         THE LOVE WAVE
C      DISPERSION FUNCTION
C
       FUNCTION FLOVE(C)
       IMPLICIT DOUBLE PRECISION(A-H, O-Z)
       DIMENSION THKNES (50), BETA(50), RHO(50)
       COMMON/NP/N, PREA/TO/THKNES, BETA, RHO,
          OMEGA
       BETNSQ=BETA(N)**2
       CSQ=C**2
       IF(BETNSQ.EQ.0.0)GOTO 130
       RBNMN=-DQRT(1.0-CSQ/BETNSQ)*BETNSQ*
          RHO(N)
       GO TO 140
   130 RBNMN=0.0
   140 XK=OMEGA/C
       C11=RBNMN
       C12=-1.0
       N2=N-2
       IF(N2.EQ.0) GO TO 175
       DO 170 J=1, N2
       M=N-J
       BETMSQ=BETA(M)**2
       ARGBTM=1.0-CSQ/BETMSQ
       IF(ARGBTM.GT.0.0) GO TO 150
       IF(ARGBTM.EQ.0.0) GO TO 170
       RBETAM=DSQRT(-ARGBTM)
       QM=XK*RBETAM*THKNES(M)
       RBMMM=RBETAM*BETMSQ*RHO(M)
       SINQM=DSIN(QM)
```

```
      B11=DCOS(QM)
      B21=RBMMM*SINQM
      GO  TO 160
150   RBETAM=-DSRT(ARGBTM)
      RBMMM=RBETAM*BETMSQ*RHO(M)
      EXPPQM=0.5*DEXP(XK*RBETAM*THKNES(M))
      EXPMQM=0.25/EXPPQM
      SINQM=EXPPQM-EXPMQM
      B11=EXPPQM+EXPMQM
      B21=-RBMMM*SINQM
160   TEMP=C11*B11-C12*B21
      C12=C11*SINQM/RBMMM+C12*B11
      C11=TEMP
170   CONTINUE
175   BETMSQ=BETA(1)**2
      ARGBTM=1.0-CSQ/BETMSQ
      IF(ARGBTM.GT.0.0)  GO  TO 180
      IF(ARGBTM.EQ.0.0)  GO  TO 200
      RBETAM=DSQRT(-ARGBTM)
      QM=XK*RBETAM*THKNES(1)
      B11=DCOS(QM)
      B21=RBETAM*BETMSQ*RHO(1)*DSIN(QM)
      GO  TO 190
180   RBETAM=-DSQRT(ARGBTM)
      EXPPQM=0.5*DEXP(XK*RBETAM*THKNES(1))
      EXPMQM=0.25/EXPPQM
      B11=EXPPQM+EXPMQM
      B21=-RBETAM*BETMSQ*RHO(1)*(EXPPQM
         -EXPMQM)
190   FLOVE=C11*B11-C12*B21
      GO  TO 210
200   FLOVE=C11
```

```
210 CONTINUE
    RETURN
    END
```

## （二） Rayleigh波频散曲线正演程序

本程序用于计算底层为固体的多层介质模型 Rayleigh 波的相速度 $c$ 及群速度 $U$。对大陆模型及海洋型两类频散曲线均可计算。

**（一） 主要标识符及子程序说明**

THKNES(50)——各层介质厚度 $d_m$ 的数组。

ALPHA(50)——各层介质纵波速度 $\alpha_m$ 的数组。

BETA(50)——各层介质横波速度 $\beta_m$ 的数组。

RHO(50)——各层介质密度 $\rho_m$ 的数组。

T(30)——计算频散曲线所取周期 $T$ 的数组。

VPH(30)——计算相速度 $c$ 的数组。

VGR(30)——计算群速度 $U$ 的数组。

THKSEA——海洋层平均厚度 $d_0$；若 THKSRA = 0，则计算大陆型频散曲线。

RHO——海水平均密度 $\rho_0$，取值1.03。

ALPSEA——海水纵波速度 $\alpha_0$，取值1.52。

N——总层数。

PREA——求相速度 $c$ 的初值。

DCDT——表示 $\partial c/\partial T$，求群速度时用。

OMEGA——圆频率 $\omega = 2\pi/T$。

ROOT($H,E,RT$)——求频散方程的根的子程序，意义与附录二同。

FRAYL($c$)——计算 Rayleigh 波频散函数 $F_R(\omega, c)$ 的子程序，其中虚元 $c$ 待求的相速度。这一程序中用到的主要符号意义为：

ALPMSQ(m) = $\alpha_m^2$,

BETMSQ(m) = $\beta_m^2$,

ALPNSQ = $\alpha_n^2$,

BETNSQ = $\beta_n^2$,

EPSQ(m) = $\rho_{m+1}/\rho_m$,

EPSQQ(m) = $2(\beta_m^2 - \mathrm{EPSQ}(m)\beta_{m+1}^2)$,

EPSILO = $(-1)^{n-1}\rho_1^2/2\beta_n^2\rho_n^2\alpha_n^2$。

其它符号可参见表8-4-1及(8-4.22)及(8-4.23)式。

（二）**FORTRAN-77语言源程序**

```
C       THIS PROGRAM IS DESIGNED TO COMPUTE
C       THE DISPERSION
C       CURVES FOR RAYLEIGH WAVE IN
C       MULTILAYERED FLAT-STRUCTURE
C
        IMPLICIT DOUBLE PRECISION (A-H, O-Z)
        DIMENSION THKNES(50), ALPHA(50), BETA(50),
           RHO(50), T(30),
           VPH(30), VGR(30)
        COMMON/NP/N, PREA/TA/THKNES, ALPHA,
           BETA, RHO, OMEGA,
           THKSEA, RHOSEA, ALPSEA
        OPEN(1, FILE='PERIOD')
        READ(1, *)(T(I), I=1, 30)
   1000 PAUSE'INPUT N, THKSEA'
        READ*, N, THKSEA
        OPEN(2, FILE='TABR')
        READ(2, *)(THKNES(I), ALPHA(I), BETA(I),
           RHO(I), I=1, N)
```

```
      RHOSEA=1.03
      ALPSEA=1.52
      IF(THKSEA.EQ.0.0)  GO  TO 300
      WRITE(*, 150)
 150  FORMT(//8X,'COMPUTATION OF THE THEORET
      IC VALUE FOR RAYLEIGH
          WAVE DISPESION IN OCENIC AREA'//10X,
          'THICKNSS', 10X,
          'SHEAR WAVE VELOCITY',10X, 'COMPRESSI
          ONAL WAVE VELOCITY',
          10X, 'DENSITY'/)
      WRITE(*, 250)THKSEA, ALPSEA, RHOSEA
 250  FORMAT(11X, F7.1, 52X, F6.2, 20X, F6.2)
      GO  TO 400
 300  WRITE(*, 310)
 310  FORMAT(//8X, 'COMPUTATION OF THE
          THERORETIC VALUE FOR RAYLEIGH
          WAVE DISPERSION'//10X, 'THEICKNESS',
          10X, 'SHEAR WAVE VELOCITY',
          10X, 'COMPRESSIONAL WAVE VELOCITY',
          10X, 'DENSITY'/)
      WRITE(*, 350)(THKNES(I), BETA(I), ALPHA)
          (I), RHO(I), I=1, N)
 350  FORMAT(11X, F7.1, 19X, F6.2,27X,F6.2,20X,F6.2)
 400  WRITE(*, 500)
 500  FORMAT(15X,'PERIOD',8X,'PHASE VELOCIT
          Y', 12X,
          'GROUP VELOCITY', /)
      IF(THKSEA.EQ.0.0)  GO  TO 20
      PREA=1.52
      GO  TO 30
```

```
20  PREA=0.9*BETA(1)
30  DO200 I=1, 30
    OMEGA=6.283185307/T(I)
    CALL ROOT(0.05, 1E-8, VPH(I))
    OMEGA=6.283185307/(T(I)-0.01)
    CALL ROOT (0.05, 1E-8, VPHA)
    OMEGA=6.283185307/(T(I)+0.01)
    CALL ROOT(0.05, 1E-8, VPHB)
    DCDT=(VPHB-VPHA)/0.02
    VGR(I)=VPH(I)/(1+T(I)/VPH(I)*DCDT)
    PREA=VPH(I)-0.1
200 CONTINUE
    WRITE(*, 100)(T(I), VPH(I), VGR(I), I=1, 30)
100 FORMAT(10X, F10.1, 9X, F12.9, 14X, F12.9)
    GO TO 100
    END
C
C      THIS IS SUBPROGRAM FOR SEEK THE ROOTS
C      OF DISPERSION EQUATION
C
    SUBROUTINE ROOT(H, E, RT)
    IMPLICIT DOUBLE PRECISON (A-H, O-Z)
    COMMON/NP/N, PREA
    W=0.0
    X=PREA
    FO=FRAYL(X)
4   X=X+H
    F1=FRAYL(X)
    IF(F1/F0.GT.0.0) GO TO 101
    FA=F0
    FB=F1
    A=X-H
```

```
      B=X
  1   C=B-(A-B)*FB/(FA-FB)
      IF(W)5, 7, 5
  7   W=1.0
      GO  TO 6
  5   IF(DABS(C1-C).LT.E)  GO  TO 3
  6   C1=C
      FC=FRAYL(C)
      IF(FC/FA.GT.0.0)  GO  TO 2
      FB=FC
      B=C
      GO  TO 1
  2   FA=FC
      A=C
      GO  TO 1
  3   RT=C
      GO  TO 102
101   F0=F1
      IF(X-10.0)4, 102, 102
102   RETURN
      END
C
C     THIS IS SUBPROGRAM FOR COMPUTATION OF
C     THE RAYLEIGH WAVE DISPESION FUNCTION
C
      FUNCTION FRAYL(C)
      IMPLICIT DONBLE PRECISION(A-H, K, L,
         O-Z)REAL KKNP, LKNP
      DIMNSION THKNES(50), ALPHA(50), BETA(50),
         RHO (50), EPSO(50),
         EPSOO(50), BETMSQ(50), ALPMSQ(50)
      COMMON/NP/N,PREA/TA/THKNES,ALPHA,BETA,
```

```
     RHO, OMEGA, THKSEA,
     RHOSEA, ALPSEA
    CSQ=C* *2
    XK=OMEGA/C
    SEACOF=0.0
    IF(THKSEA.EQ.0.0) GO TO 160
    RALPHO=DSQRT(CSQ/ALPSEA* *2-1.0)
    SEAARG=XK*RALPHO*THKSEA
    SEACOF=RHOSEA*TAN(SEAARG)/(RALPHO*
     RHO(1))
160 GAMMA1=2*BETA(1)* *2/CSQ
    GAM1M1=GAMMA1-1.0
    UKNP=-GAMMA1*GAM1M1
    VKNP=0.0
    WKNP=GAM1M1* *2
    RKNP=GAMMA1* *2
    SKNP=SEACOF
    ALPNSQ=ALPHA(N)* *2
    BETNSQ=BETA(N)* *2
    RHONSQ=RHO(N)* *2
    EPSILO=(-1)* *(N-1)*RHO(1)* *2/(2*ALPNSQ
     *RHONSQ*BETNSQ)
    N1=N-1
170 DO 230 M=1, N1
    BETMSQ(M)=BETA(M)* *2
    EPSO(M)=RHO(M+1)/RHO(M)
    EPSOO(M)=2*(BETMSQ(M)-EPSO(M)*BETA(M
     +1)* *2)
    EPS15=-EPSO(M)
    EPS1=EPSOO(M)/CSQ
    EPS2=EPS1-1.0
    EPS3=EPS1-EPS15
```

```
      EPS4 =EPS2-EPS15
      THKKM =THKNES(M) * XK
      ALPMSQ(M) =ALPHA(M) * * 2
      ARGALM =1.0-CSQ/ALPMSQ(M)
      IF(ARGALM.GE.0.0)  GO  TO 190
      RALPHM =DSQRT(-ARGALM)
      PM =THKKM * RALPHM
      SINPM =DSIN(PM)
      ZETA1 =DCOS(PM)
      ZETA3 =RALPHM * SINPM
180   ARGBTM =1.0-CSQ/BETMSQ(M)
      IF(ARGBTM.GE.0.0)  GO  TO 200
      RBTAM =DSQRT(-ARGBTM)
      QM =THKKM * RBETAM
      SINQM =DSIN(QM)
      ZETA2 =DCOS(QM)
      ZETA5 =RBETAM * SINQM
      GO  TO 210
190   RALPHM = - DSQRT(ARGALM)
      EXPPPM =0.5 * DEXP(THKKM * RALPHM)
      EXPMPM =0.25/EXPPPM
      SINPM =EXPPPM-EXPMPM
      ZETA1 =EXPPPM+EXPMPM
      ZETA3 = - RALPHM * SINPM
      GO  TO 180
200   RBETAM = - DSQRT(ARGBTM)
      EXPPQM =0.5 * DEXP(THKKM * RBETAM)
      EXPMQM =0.25/EXPPQM
      SINQM =EXPPQM-EXPMQM
      ZETA2 =EXPPQM+EXPMQM
      ZETA5 = - RBETAM * SINQM
210   ZETA4 =SINPM/RALPHM
```

```
      ZETA6 = SINQM/RBETAM
      ZETA7 = ZETA1 * ZETA2
      ZETA8 = ZETA1 * ZETA5
      ZETA9 = ZETA1 * ZETA6
      UKN = 2.0 * UKNP
      VKN = VKNP
      IF((2 * (M/2)).EQ.M) GOTO 220
      XKNP = ZETA4 * (ZETA2 * VKNP + ZETA6 * WKNP)
         - ZETA7 * RKNP + ZETA9 * SKNP
      ZKNP = ZETA8 * VKNP - ZETA7 * WKNP + ZETA3
         * (ZETA5 * RKNP + ZETA2 * SKNP)
      UKNP = - (EPS1 * EPS4 + EPS2 * EPS3) * UKNP
         + EPS2 * EPS4 * XKNP + EPS1 * EPS3 * ZKNP
      VKNP = EPS15 * (ZETA4 * (ZETA * VKNP - ZETA2
         * WKNP( - ZETA8 * RKNP - ZETA7 * SKNP)
      SKNP = EPS15 * ( - ZETA7 * VKN - ZETA9 * WKNP
         - ZETA3 * (ZETA2 * RKNP - ZETA6 * SKNP))
      WKNP = EPS2 * (EPS2 * XKNP - EPS1 * UKN)
         + EPS1 * EPS1 * ZKNP
      RKNP = EPS4 * (EPS4 * XKNP - EPS3 * UKN)
         + EPS3 * EPS3 * ZKNP
      GO  TO 230
220   KKNP = ZETA9 * VKNP + ZETA7 * WKNP - ZETA4
         * (ZETA6 * RKNP - ZETA2 * SKNP)
      LKNP = ZETA3 * (ZETA2 * VKNP - ZETA5 * WKNP)
         + ZETA7 * RKNP + ZETA8 * SKNP
      UKNP = - (EPS1 * EPS4 + EPS2 * EPS3) * UKNP
         + EPS2 * EPS4 * KKNP + EPS1 * EPS3*LKNP
      VKNP = EPS15*(ZETA3*(ZETA6*VKNP + ZETA2
         *WKNP)ZETA9*RKNP - ZETA7*SKNP)
      SKNP = EPS15*( - ZETA7*VKN + ZETA8*WKNP
         + ZETA4*(ZETA2*RKNP + ZETA5*SKNP))
```

```
      WKNP=EPS4*(-EPS4*KKNP+EPS3*UKN)
         -EPS3*EPS3*LKNP
      RKNP=EPS2*(-EPS2*KKNP+EPS1*UKN)
         -EPS1*EPS1*LKNP
230   CONTINUE
240   RALPHN=-DSQRT(1.0-CSQ/ALPNSQ)
      RBETAN=-DSQRT(1.0-CSQ/BETNSQ)
      RALRBT=RALPHN*RBETAN
      EPSILN=-EPSILO*CSQ**2/RALRBT
      IF((2*(N/2).EQ.N))GOTO 250
      FRAYL=EPSILN*(-VKNP*RBETAN+WKNP
         -RKNP*RALRBT-SKNP*RALPHN)
      GOTO 260
250   FRAYL=EPSILN*(VKNP*RALPHN+RKNP)
         -WKNP*RALRBT+SKNP*RBETAN)
260   RETURN
      END
```

## （三）Rayleigh波频散曲线反演程序

下面，我们给出一个用 Rayleigh 面波频散曲线反演地下介质分层结构的完整程序。这个程序要求输入实际观测的 Rayleigh 波频散曲线，并输入地下地壳及地幔结构的初始模型，以及由约束条件所限制的待反演参数的信息。

这个程序具有通用性，即既可计算群速度反演问题，也可计算相速度反演问题。此外，亦可用于计算正演问题。只要作适当的变动，还可以计算 Love 波频散曲线的正演及反演问题。

（一）主要标识符及子程序说明

1。主程序

FP(20)、FR(20)——两个配合使用的数组，表示岩石密度及纵波速度的统计关系。其中 FP表示

纵波速度值，$FR$表示对应的密度值。

T(30)——实测或理论频散曲线的周期值。

QBDIS(30)——相速度或群速度观测值。

F(30)——相速度或群速度理论计算值。

BETA(20)——作正演计算时为层状介质为各层的横波速度；作反演计算时为相应的初值。

LX(40)，BT(40)——这是两个配合使用的数组。作反演计算时，$BT$数组中分别按$\beta_1$，$h_1$，$\beta_2$，$h_2$，…，$\beta_L$，$h_L$顺序将各层的横波速度及厚度存放入内。LX数组为对应的标志数组，若对应于BT数组中某一变量的值为0，表示此变量在反演过程中变化。若取为1，则对应变量在反演过程中不变化(固定约束)。

ZQNE(40)——由BT数组中挑选出来排列的$\beta$及$h$值数组。先为初始值，反演结束后为计算值。

ZONEO(40)——由BT数组中挑选出来的$\beta$及$h$数组。一直保留不变，最后作模型的初值输出。

KPG——标志作相速度或群速度计算的整型变量。KPG = 0表示相速度的正演或反演计算。KPG = 1表示群速度的正演或反演计算。

KK——标志作正演或反演计算的整型变量。KK = 0为正演，KK = 1为反演。

L——地壳及地幔总的分层数。

M——频散曲线的实际观测点数。

N——反演计算中待变化的参数数目。

EA——反演计算中目标函数$\Phi$的精度要求。

BL，BU——地下介质平均速度$\overline{\beta}$的限制范围，BL为$\overline{\beta}$的下限，BU为$\overline{\beta}$的上限。即BL$\leqslant\overline{\beta}\leqslant$BU。

HL，HU——分别表示地下分层介质总厚度 $H_m$ 的限制范围，HL 为下限，HU 为上限。即 HL $\leqslant H_m \leqslant$ HU。

2. 子程序 。DISRAY(X)

计算 Rayleigh 波频散曲线（相速度或群速度）的子程序。其中 X 表示层参数 $\beta_i$ 及 $h_i$。这一子程序中的主要参数如下：

ALPHA(20)——各层介质的纵波速度 $\alpha$ 值，一般按关系式 $\alpha=Y\beta$ 计算，由横波速度 $\beta$ 求得。

RHO(20)——各层介质的密度 $\rho$ 值，由密度 $\rho$ 与速度 $\alpha$ 关系曲线内插求得。

VPH(30)——计算的相速度理论值。

VGR(30)——计算的群速度理论值。

QMEGA——表示圆周率 $\omega=2\pi/T$。

PREA——用线性函数迭代求根的初值。

JP——控制变量，作反演计算时若 JP=0 仅输出中间计算结果。

3. 子程序 CONFINE(X)

反演过程中约束各层介质参数的子程序。X 为各层横波速度 $\beta_i$ 及层厚度 $h_i$。

约束条件为上地幔横波速度限制为 $4.4\leqslant\beta\leqslant4.75$，地壳中各层的横波速度限制为 $2.5\leqslant\beta_i\leqslant4.9$。有关的参数表示为：

HSUM——地壳总厚度 $\sum h_i$，根据具体地区选择约束数值。

BEAN——地壳中横波速度的权平均值 $\overline{\beta}=\left(\sum_{i=1}^{L}\beta_i h_i\right)\Big/\sum_{i=1}^{L}h_i$，由计算者自行选择。

4. 子程序 MAQT(X，PLMDO，KENN，S，ITFR)

阻尼最小二乘法子程序，用于求目标函数 $\Phi$ 的极小。其中 X 开始表示层参数 $\beta_i$ 和 $h_i$ 的初值，计算结束时为修定后的层参数值。

PLMDO——阻尼因子初值。

KENN——反演迭代过程终止的特征整型参量，有下列四种情况：

KENN = 0 迭代过程在满足精度要求时结束；

KENN = 1 迭代过程已达到最大次数（程序中规定30次）而结束；

KENN = −1 标志对λ搜索 $S(X+\lambda\Delta X)$ 极小的过程已失效；

KENN = −2 标志解求修正量 $\Delta X$ 的线性方程组失败。

S—— 反演结束时残差平方和。

ITER——迭代次数。

其中各参数的意义是：

AC(30，40)——Jacobi 矩阵数组。

AA(40，40)——正则线性方程组系数矩阵。

G(40)——工作单元，表示正则方程组的右端项。

DX(40)——反演过程中参数的修正量 ΔX。

QLDX(40)——反演过程中参数前一次迭代值。

DEF(30)——实测相速度(或群速度)与理论计算值之差。

5. 子程序 DERIVE(X，AC，N1)

计算 Jacobi 矩阵 AC 的子程序。X 为层参数，N1 为 Jacob 矩阵的列数(即待反演的参数个数)。

6. 子程序 LDLT(ISW)

用 CHOLESKY 方法解系数矩阵为正定对称的线性方程组的程序。ISW = 0 表示方程无解，ISW = 1 表示方程有解，

7. 子程序 QIP(N，X，Y，U，F)

一元三点插值程序。N 为结点数，X 为结点自变量数组，Y 为结点函数数组，U 为欲内插的自变量值，F 为待求函数值。

8. 子程序 ROOT(H，E，RT)

求根子程序，意义同附录二。

9. 子程序 FRAYL(C)

计算 Rayleigh 波频散函数 $F_L(\omega, c)$ 程序。

10. 子程序 CALSRC(C)

计算协方差矩阵及分辨矩阵的程序。其中 C(40,40) 开始为正则方程系数矩阵$A^TA$，结束时为协方差矩阵。

BB(40, 40)——工作单元，$(A^TA+\theta^2I)^{-1}$；

R(40, 40)——分辨矩阵；

DDX(40)——各反演待求参数的标准偏差值。

11. 子程序 GJ1(N, A, ISW)

对称矩阵求逆的子程序。其中 $N$ 为对称矩阵 $A$ 的阶数；ISW = 0 求逆失败，ISW = 1 求逆成功。

（二） FORTRAN-77 语言源程序

```
C        THIS PROGRAM IS DESIGNED FOR
C        COMPUTATION OF THE INVERSE
C        OR FORWARD PROBLEM OF RAYLEIGH WAVE
C        DISPERION IN PHASE
C        VELOCITY OR GROUP VELOCITY

         IMPLICIT DOUBLE PRECISION(A-H, O-Z)
         DIMENSION FP(20), FR(20), T(30), OBDIS(30),
            F(30), THKNES(20),
            BETA(20),ZONEO(40),ZONE(40),LX(40), BT(40)
         COMMON/K/KK, KPG/NM/N, M/FPR/FP, FR/OBD/
            /OBDIS/TF/T, F/LB/LX, BT/TB/
            THKNES,BETA/BH/BL,BU,HL, HU/E/EA/L/L,
            NL/SEA/THKSEA, RHEOSA, ALPSEA
         OPEN(8, FILE='FPR')
         READ(8, *)(FP(I), FR(I), I=1, 20)
   1000  PAUSE 'INPUT KK,KPG, L,M,EA,BL,BU,HL,HU'
         READ*, KK, KPG, L, M, EA, BL, BU, HL, BU
```

```
1111 OPEN(1, FILE='THBE')
     READ(1, *)(THKNES(I), BETA(I), I=1, L),
        THKSEA
     DO 100 I=1, L
     BT(2*I-1)=BETA(I)
     ZONE(2*I-1)=BETA(I)
     ZONEO(2*I-1)=BETA(I)
     BT(2*I)=THKNES(I)/10
     ZONE(2*I)=BT(2*I)
     ZONEO(2*I)=BT(2*I)
 100 CONTINUE
     IF(KK.NE.0) GO TO 111
     CALL DISRAY (ZONEO)
     PAUSE 1111
     GO TO 1111
 111 OPEN(2, FILE='OBD')
     READ(2, *)(OBDIS(I), I=1, M)
     NL=2*L
     OPEN(3, FILE='LX')
     READ(3, *)(LX(I), I=1, NL)
     DO 7 I=1, NL
     IF(LX(I).NE.0) GO TO 7
     J=J+1
     ZONE(J)=BT(I)
     ZONEO(J)=BT(I)
   7 CONTINUE
     N=J
     WRITE(*, 888)BL, BU, HL, HU
 888 FORMAT(//50X, 'INVERSION FOR RAYLEIGH
        WAVE DISPERSION'//20X, 'ITERATIVE
        PROCESS AS FOLLOWS'//20X,'BETA(L-U)',
        2F10.2, 10X, 'THKNES(L-U)', 2F10.2)
```

```
      WRITE(*, 20)M, (T(I), I=1, M)
 20   FORMAT(/10X, 'M=', I4/5, X'T=', 30F6.2)
      WRITE(* 30)(OBDIS(I), I=1, M)
 30   FORMAT('OBDIS', 30F6.2)
      WRITE(*, 50)(ZONE(I), I=1, N)
 50   FORMAT('X=', 30F6.2)
      CALL MAQT(ZONE, 1.0, KENN, S, ITER)
      WRITE(*, 444)
444   FORMAT(//20X, 'THIS IS  COMPUTED
         PARAMETERS'/)
      WRITE(*, 333)KENN, S, ITER
333   FORMAT(/20X, 'KENN=', I2, 5X, 'S=',
         F9.7, 5X, 'ITER=', I3/)
      WRITE(*, 200)
200   FORMAT(/14X, 'PERIOD', 8X, 'THEORETIC
         VALUE', 8X, 'OBSERVED VALUE'/)
      WRITE(*, 555)(T(I), F(I), OBDIS(I), I=1, M)
555   FORMAT(10X, F9.2, 10X, F10.3, 10X, F10.3)
      WRITE(*, 300)
300   FORMAT(/15X, 'INVERSED VALUE', 10X,
         'PRELIMINARY VALUE')
      WRITE(*, 666)(ZONE(I), ZONEO(I), I=1, N)
666   FORMAT(/20X, F6.2, 20X, F6.2)
222   PAUSE 1000
      GO  TO 1000
      END
C
C     THIS IS SUBPROGRAM FOR COMPUTAION OF
C     DISPERSION CURVES OF RAYLEIGH WAVE
C
      SUBROUTINE DISRAY(X)
      IMPLICIT DOUBLE PRECISION(A-H, O-Z)
```

```
      DIMENSION X(40),ALPHA(20),BETA(20),RHO(20)
         THKNES(20), T(30), FP(20), FR(20), F(30),
         VPH(30), VGR(30)
      COMMON/K/KK, KPG/NM/N, M/TB/THKNES,
         BETA/AR/
         ALPHA, RHO/TF/T, F/FPR/FP, FR/L/L, NL
         /P/PREA/JJ/JP/OM/OMEGA/SEA/THKSEA,
         RHOSEA, ALPSEA
      IF(KK.EQ.0)  GO  TO 15
      CALL CONFIN(X)
 15   G=1.73
      DO 101 I=1, L
      ALPHA(I)=G*BETA(I)
101   CALL QIP(20, FP, FR, BETA(I), RHO(I))
      RHOSEA=1.03
      ALPSEA=1.52
      IF(THKSEA.EQ.0.0) GO  TO 20
      PREA=1.52
      GO  TO 30
 20   PREA=0.9*BETA(1)
 30   IF(KK*JP.EQ.0) GO  TO 600
      HM=0.0
      L1=L-1
      DO 40  I=1, L1
 40   HM=HM+THKNES(I)
      WRITE(*, 50)HM, (THKNES(I), I=1, L)
 50   FORMAT(/10X, 'HSUM=', F7.2, 5X, 20F7.2)
      WRITE(*, 60)(BETA(I), I=1, L)
 60   FORMAT(/10X, 'BETA=', 5X, 20F7.2)
      WRITE(*, 70)(ALPHA(I), I=1, L)
 70   FORMAT(/10X, 'ALPHA=', 5X, 20F7.2)
      WRITE(*, 80)(RHO(I), I=1, L)
```

```
80   FORMAT(/10X, 'RHO=', 5X, 20F7.4)
600  DO 620 I=1, M
     OMEGA=6.283185307/T(I)
     BT1=BETA(L)-0.1
     SH=0.05
     IF(PREA.GE.BT1)SH=0.005
     CALL ROOT(SH, 1E-6, F(I))
     VPH(I)=F(I)
     IF(KPG.EQ.0) GO TO 620
     OMEGA=6.283185307/(T(I)-0.01)
     CALL ROOT(SH, 1E-6, VPHA)
     OMEGA=6.283185307/(T(I)+0.01)
     CALL ROOT(SH, 1E-6, VPHB)
     DCDT=(VPHB-VPHA)/0.02
     VGR(I)=F(I)/(1+T(I)/F(I)*DCDT)
     F(I)=VGR(I)
     PREA=F(I)-0.1
620  CONTINUE
     IF(KK.NE.0) GO TO 800
     IF(THKSEA.EQ.0) GO TO 300
     WRITE(*, 150)
150  FORMAT(//8X,'COMPUTAION OF THE THEORE-
        TIC
        VALUE FOR RAYLEIGH WAVE DISPERSION
        INOCENIC AREA'//10X, 'THICKNESS', 10X,
        'SHEAR WAVE VELOCITY',10X,'COMPRESSIO-
        N·AL
        WAVE VELOCITY', 10X, 'DENSITY'/)
     WRITE(*, 250)THKSEA, ALPSEA, RHOSEA
250  FORMAT(11X, F7.1, 52X, F6.2, 20X, F6.2)
     GO TO 400
300  WRITE(*, 320)
```

```
320  FORMAT(//8X, 'COMPUTATION OF THE
        THEORETIC
        VALUE FOR RAYL EIGH WAVE DISPERSIO
        N'//10X, 'THEICKNESS', 10X, 'SHEAR WAVE
        VELOCITY', 10X, 'COMPRESSIONAL WAVE
        VELOCITY', 10X, 'DENSITY'/)
     WRITE(*, 350)(THKNES(I), BETA(I), ALPHA
        (I), RHO(I), I=1, L)
350  FORMAT(11X, F7.1, 19X, F6.2, 27X, F6.2, 20X,
        F6.2)
400  WRITE(*, 500)
500  FORMAT(15X,'PERIOD',8X',PHASE VELOCITY',
      12X,'GROUP VELOCITY',8X,'GROUP VELOCITY'
     WRITE(*, 100)(T(I), VPH(I), VGR(I), I=1,
        M)/)
100  FORMAT(10X,F10.1, 9X, F12.9, 14X, F12.9)
800  RETURN
     END
C
C     THIS IS SUBPROGRAM FOR PARAMETERS
C     CONSTRAINED
C     IN PROCESS OF INVERSE COMPUTAION
C
     SUBROUTINE CONFIN(X)
     IMPLICIT DOUBLE PRECISION(A-H, O-Z)
     DIMENSION X(40), BETA(20), THKES(20), LX
        (40), BT(40)
     COMMON/TB/THKNES, BETA/LB/LX, BT/L/L, NL
        /BH/BL, BU, HL, HU/N/N, M/JJ/JP
     J=0
     DO 7 I=1, NL
     IF(LX(I), NE, 0)  GO TO 7
```

```
      J = J + 1
      BT(I) = X(J)
    7 CONTINUE
      WRITE(*, 17)(BT(I), I = 1, NL)
   17 FORMAT(10X, 'BE(I), TH(I)', 7X, 20F8.4)
   18 DO 8 I = 1, L
      BETA(I) = BT(2 * I - 1)
    8 THKNES(I) = BT(2 * I) * 10.0
      IF(BETA(L).LT.4.4)BETA(L) = 4.4
      IF(BETA(L).GT.4.75)BETA(L) = 4.75
      L1 = L - 1
      DO 9 I = 1, L1
      IF(BETA(I).LT.2.5)BETA(I) = 2.5
      IF(BETA(I).GT.4.9)BETA(I) = 4.9
    9 IF(BETA(I).LT.BETA(1))BETA(I) = BETA(1)
   10 BMAX = 0.0
      BMIN = 1E5
      DO 51 I = 2, L1
      IF(BETA(I).LT.BMAX) GO TO 50
      BMAX = BETA(I)
      KBMAX = I
   50 IF(BETA(I).GT.BMIN) GO TO 51
      BMIN = BETA(I)
      KBMIN = I
   51 CONTINUE
   20 HMAX = 0.0
      HMIN = 1E5
      BEAN = 0.0
      HSUM = 0.0
      DO 54 I = 1, L1
      IF(THKNES(I).LT.HMAX) GO TO 52
      HMAX = THKNES(I)
```

```
      KHMAX = I
52    IF(THKNES(I).GT.HMIN)  GO  TO 53
      HMIN = THKNES(I)
      KHMIN = I
53    BEAN = BEAN + BETA(I) * THKNES(I)
      HSUM = HSUM + THKNES(I)
54    CONTINUE
      BEAN = BEAN/HSUM
60    IF(HSUM.LT.HU) GO  TO 61
      THKNES(KHMAX) = THKNES)KHMAX) - 20
      GO  TO 20
61    IF(HSUM.GT.HL)  GO  TO 62
      THKNES(KHMIN) = THKNES(KHMIN) + 2.0
      GO  TO 20
62    IF(BEAN.LT.BU) GO  TO 63
      BETA(KBMAX) = BETA(KBMAX) - 0.01
      GO  TO 10
63      IF(BEAN.GT.BL) GO  TO 64
      BETA(KBMIN) = BETA(KBMIN) + 0.1
      GO  TO 10
64    DO 80  I = 1, L
      BT(2 * I - 1) = BETA(I)
80    BT(2 * I) = THKNES(I)/10.0
      J = 0
      DO 90  I = 1, NL
      IF(LX(I)NE0)  GO  TO 90
      J = J + 1
      X(J) = BT(I)
90    CONTINUE
111   RETURN
      END
C
```

```
C       THIS IS SUBROUTINE OF DAMPING LEAST
C       SQUARES
C       METHOD TO SEARCH THE MINIMUM OF OBJ-
C       ECT
C       FUNCTION IN INVERSE COMPUTATION
C
        SUBROUTINE MAQT(X, PLMDO, KENNS, ITER)
        IMPLICIT DOUBLE PRECISION (A-H, O-Z)
        DIMENSION X(40), T(30), F(30), AC(30, 40),
            AA(40, 40), G(40),
            DX(40), OLDX(40), DEF(30)
        COMMON/NM/N, M/TF/T, F/ACC/AC/A/AA/DE/
            DEF/GG/G
            /D/DX/E/EA/JJ/JP/PL/PLMD/S/SS
        WRITE(*, 5)
      5 FORMAT(/20X, 'DAMPING LEAST SQUARES
            METHOD')
        ITER=0
        PLMD=PLMDO
        JP=2
        ISB=1
        KENN=0
        GO  TO 1000
   1111 PHIO=PHI
     10 ITER=ITER+1
        WRITE(*, 11)
     11 FORMAT(10X,28('* * * * *'))
        WRITE(*, 12)ITER, PLMD
     12 FORMAT(/40X, 'ITER=', I3, 10X, 'PLMD=',
            F8.4)
        IF(ITER.LE.30)  GO  TO 30
        KENN=1
```

```
      GO TO 10000
   30 CALL DERIVE(X, AC, N)
      DO 60 I=1, N
      HF=0.0
      DO 70 K=1, M
   70 HF=HF+AC(K, I)*DEF(K)
      G(I)=HF
      DX(I)=HF
      DO 80 K=I, N
      HF=0.0
      DO 90 J=1, M
   90 HF=HF+AC(J, I)*AC(J, K)
      AA(I, K)=HF
   80 AA(K, I)=HF
   60 AA(I, I)=AA(I, I)+PLMD
      CALL LDLT(ISW)
      IF(ISW.EQ.1) GO TO 100
      KENN=-2
      GO TO 10000
  100 DO 120 I=1, N
      OLDX(I)=X(I)
  120 X(I)=X(I)+DX(I)
      ISB=2
      GO TO 1000
 2222 WRITE(*, 22) PHI, PHIO
   22 FORMAT(20X, 'PHINEW=', F14.9, 5X, 'PHIOLD
         =', F14.9)
      IF(PHI.GE.PHIO) GO TO 130
      C=1.0
      GO TO 140
  130 GDX=0.0
      DO 160 I=1, N
```

```
160  GDX = GDX + G(I) * DX(I)
170  L = 0 C = GDX/(PHI - PHIO + 20*GDX)
180  L = L + 1
     DO 190 I = 1, N
190  X(I) = OLDX(I) + C * DX(I)
     ISB = 3
      GO  TO 1000
3333 IF(PHI.LT.PHIO) GO  TO 140
     IF(L.LT.2)  GO  TO 210
     DO 220 I = 1, N
220  X(I) = OLDX(I)
     PLMD = 1.5 * PLMD
     GO  TO 10
210  C = O.5 * C
     GO  TO 180
140  HF = 0.0
     HZ = 0.0
     DO 230 I = 1, N
     HF = HF + ABS(C * DX(I))
230  HZ = HZ + ABC(X(I))
     IF(HF.LT.1E - 3 * HZ) GO  TO 10000
     IF(PHI.LT.EA)  GO  TO 10000
     PHIO = PHI
     PLMD = 0.5 * PLMD
     GO  TO 10
1000 CALL DISRAY(X)
     CALL CALPHI(PHI)
     WRITE(* , 6000) PHI, (X(I), I = 1, N)
6000 FORMAT(8X, 'PHI = ', F10.5, 2(/16X, 'X = ',
        13F12.9))
     WRITE(* , 5000)(F(I), I = 1, M)
```

```
 5000  FORMAT(3(/10F14·9))
       GO TO (1111, 2222, 3333), ISB
10000  S=PHI
       SS=S
       CALL DERIVE(X, AC, N)
       DO 900 I=1, N
       DO 800 K=1, N
       HF=0.0
       DO 700 J=1, M
  700  HF=HF+AC(J, I)*AC(J, K)
       AA(I, K)=HF
  800  AA(K, I)=HF
  900  CONTINUE
       CALL CALSRC(AA)
       RETURN
       END
C
       THIS IS SUBPROGRAM TO SET UP THE PARTI-
C      AL
C      DIFFERENTIAL MATRAX
C
       SUBROUTINE DERIVE(X, AC, N1)
       IMPLICIT DOUBLE PRECISION (A-H, O-Z)
       DIMENSION X(N1), AC(30,N1), T(30), F(30), TEDO
         (30)
       COMMON /NM/N, M/TF/T, F/JJ/JP
       JP=0
       CALL DISRAY(X)
       DO 300 I=1, M
  300  TEDO(I)=F(I)
       DO 100 I=1, N
       X(I)=X(I)+1E-3
```

```
      CALL DISRAY(X)
      X(I)=X(I)-1E-3
      DO 100 J=1, M
      AC(J, I)=(F(J)-TEDO(J))*1E3
100   CONTNUE
      JP=1
      RETURN
      END
C
      THIS IS SUBPROGRAM TO COMPUTE THE
C     VALUE OF
C     OBJECT FUNCTION
C
      SUBROUTINE CALPHI(PHI)
      IMPLICIT DOUBLE PRECISION(A-H, O-Z)
      DIMENSION T(30), F(30), OBDIS(30), DEF(30)
      COMMON/TF/T, F/OBD/OBDIS/DE/DEF/NM/N, M
      PHI=0.0
      DO 200 I=1, M
      DEF(I)=OBDIS(I)-F(I)
      PHI=PHI+DEF(I)*DEF(I)
200   CONTINUE
      RETURN
      END
C
C     THIS IS SUBPROGRAM OF INTERPOLATION
C
      SUBROUTINE QIP(N, X, Y, U, F)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DIMENSION X(N), Y(N)
      NM1=N-1
      DO 10 I=2, NM1
```

```
      IF(U.LE.X(I))  GO  TO 20
   10 CONTINUE
      I=NMI  GO  TO  30
   20 IF(I.NE.2)  GO  TO 25
      GO  TO 30
   25 IF(U-X(I-1).LT.X(I)-U)I=I-1
   30 X0=X(I-1)
      X1=X(I)
      X2=X(I+1)
      A1=(U-X1)*(U-X2)/((X0-X1)*(X0-X2))
      A2=(U-X2)*(U-X0)/((X1-X2)*(X1-X0))
      A2=(U-X0)*(U-X1)/((X2-X0)*(X2-X1))
      F=A1*Y(I-1)+A2*Y(I)+A3*Y(I+1)
      RETURN
      END
C
C     THIS IS SUBPROGRAM TO SOLVE LIEAR SYST
C     EM
C     BY CHOLESKY METHOD

      SUBROUTINE LDLT(ISW)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DIMENSION AA(40, 40), P(40), G(40), DX(40)
      COMMON/NM/N, M/A/AA/GG/G/D/DX
      DO 5 I=1, N
      DO 5 J =1, I
      W=AA(J, I)
      IF(I-J)10, 20, 10
   20 IF(J.EQ.1) GO  TO 30
      DO 25 L=2, J
      K=J-L+1
      Y=AA(I, K)
      Z=Y*P(K)
```

```
      AA(I, K)=Z
   25 W=W-Y*Z
   30 IF(W.NE.0.0) GO TO 35
      ISW=0
      RETURN
   35 P(I)=1.0/W
      GO TO 5
   10 IF(J.EQ.1) GO TO 45
      DO 40 L=2, J
      K=J-L+1
   40 W=W-AA(I, K)*AA(J, K)
   45 AA(I, J)=W
    5 CONTINUE
      DO 60 I=1, N
      Y=G(I)
      IF(I.EQ.1) GO TO 60
      DO 65 L=2, I
      K=I-L+1
   65 Y=Y-AA(I, K)*DX(K)
   60 DX(I)=Y
      DO 100 L=1, N
      I=N-L+1
      Y=DX(I)*P(I)
      IF(I.EQ.N) GO TO 100
      IA1=I+1
      DO 70 K=IA1, N
   70 Y=Y-AA(K, I)*DX(K)
  100 DX(I)=Y
      ISW=1
      RETURN
      END
C
```

```
C       THIS IS SUBPROGRAM FOR SEEK THE ROOTS
C       OF
C       DISPERSION EQUATIONS
C
        SUBROUTINE ROOT(H, E, RT)
        IMPLICIT DOUBLE PRECISION(A-H, O-P)
        COMMON/P/PREA
        W=0.0
        X=PREA
        F0=FRARL(X)
      4 X=X+H
        F1=FRAYL(X)
        IF(F1*F0.GT.0.0)  GO  TO 101
        FA=F0
        FB=F1
        A=X-H
        B=X
      1 C=B-(A-B)*FB/(FA-FB)
        IF(W)5, 7, 5
      7 W=1.0
        GO  TO 6
      5 IF(DABS(C1-C).LT.E)  GO  TO 3
      6 C1=C
        FC=FRAYL(C)
        IF(FC*FA.GT.0.0)  GO  TO 2
        FB=FC
        B=C
        GO  TO 1
      2 FA=FC
        A=C
        GO  TO 1
      3 RT=C
```

```
      GO  TO 102
101   F0=F1
      IF(X-10.0)4, 102, 102
102   RETURN
      END
C
C     THIS IS SUBPROGRAM FOR COMPUTATION OF
C     THE RAYLEIGH WAVE DISPERSION FUNCTION
C
      FUNCTION FRAYL(C)
      IMPLICIT DOUBLE PRECION(A-H, O-Z)
      DOUBLE PRECISION KKNP, LKNP
      DIMENSION THKNES(20), ALPHA(20),BETA(20),
         RHO(20),
         EPSO(20),EPSOO(20),BETMSQ(20),ALPMSQ(20)
      COMMON  /L/L, NL/P/PREA/TB/THKNES, BETA /
         AR/ALPHA,
        RHO/OM/OMERA/SEA/THSEA,RHOSEA,ALPSEA
      N=L
      CSQ=C* *2
      XK=OMEGA/C
      SEACOF=0.0
      I F(THKSEA.EQ.0.0)  GO  TO  160
      RALPHO=DSQRT(CSQ/ALPSEA* *2-1.0)
      SEAARG=XK*RALPHO*THKSEA
      SEACOF=RHOSEA*TAN(SEAARG)/(RALPHO*
      RHO(1))
160   GAMMA1=2*BETA(1)* *2/CSQ
      GAM1M1=GAMMA1-1.0
      UKNP=-GAMMA1-1.0
      UKNP=-GAMMA1*GAM1M1
      VKNP=0.0
```

```
      WKNP=GAM1M1* *2
      RKNP=GAMMA1* *2
      SKNP=SEACOF
      ALPNSQ=ALPHA(N)* *2
      BETNSQ=BETA(N)* *2
      RHONSQ=RHO(N)* *2
      EPSILO=(-1)* *(N-1)*RHO(1)* *2/(2*ALPN
         SQ*RHONSQ*BETNSQ)
      N1=N-1
170   DO 230 M=1, N1
      BETMSQ(M)=BETA(M)*BETA(M)
      EPSO(M)=RHO(M+1)/RHO(M)
      EPSOO(M)=2*(BETMSQ(M)-EPSO(M)*BETA(M
      +1)
      *BETA(M+1))
      EPS15=-EPSO(M)
      EPS1=EPSOO(M)/CSQ
      EPS2=EPS1-1.0
      EPS3=EPS1-EPS15
      EPS4=EPS2-EPS15
      THKKM=THKNES(M)*XK
      ALPMSQ(M)=ALPHA(M)*ALPHA(M)
      ARGALM=1.0-CSQ/ALPMSQ(M)
      IF(ARGALM.GE.0.0)  GO  TO 190
      RALPHM=DSQRT(-ARGALM)
      PM=THKKM*RALPHM
      SINPM=DSIN(PM)
      ZETA1=DCOS(PM)
      ZETA3=RALPHM*SINPM
180   ARGBTM=1.0-CSQ/BETMSQ(M)
      IF(ARGBTM.GE0.0)  GO  TO 200
      RBETAM=DSQRT(-ARGBTM)
      QM=THKKM*RBETAM
```

```
      SINQM=DSIN(QM)
      ZETA2=DCOS(QM)
      ZETA5=RBETAM*SINQM
      GO TO 210
190   RALPHM=-DSQRT(ARGALM)
      EXPPPM=0.5*DEXP(THKKM*RALPHM)
      EXPMPM=0.25/EXPPPM
      SINPM=EXPPPM-EXPMPM
      ZETA1=EXPPPM+EXPMPM
      ZETA3=-RALPHM*SINPM
      GO TO 180
200   RBETAM=-DSQRT(ARGBTM)
      EXPPQM=0.5*DEXP(THKKM*RBETAM)
      EXPMQM=0.25/EXPPQM
      SINQM=EXPPQM-EXPMQM
      ZETA2=EXPPQM+EXPMQM
      ZETA5=-RBETAM*SINQM
210   ZETA4=SINPM/RALPHM
      ZETA6=SINQM/RBETAM
      ZETA7=ZETA1*ZETA2
      ZETA8=ZETA1*ZETA5
      ZETA9=ZETA1*ZETA6
      UKN=2.0*UKNP
      VKN=VKNP
      IF((2*(M/2)).EQ.M) GO TO 220
      XKNP=ZETA4*(ZETA2*VKNP+ZETA6*
          WKNP)
          -ZETA7*RKNP+ZETA9*SKNP
      ZKNP=ZETA8*VKNP-ZETA7*WKNP+ZETA3
          *(ZETA5*RKNP+ZETA2*SKNP)
      UKNP=-(EPS1*EPS4+EPS2*EPS3)*UKNP
          +EPS2*EPS4*XKNP+EPS1*EPS3*ZKNP
```

```
      VKNP=EPS15*(ZETA4*(ZETA5*VKNP-ZETA2
         *WKNP)-ZETA8*RKNP-ZETA7*SKNP)
      SKNP=EPS15*(-ZETA7*VKN-ZETA9*WKNP
         -ZETA3*(ZETA2*RKNP-ZETA6*SKNP))
      WKNP=EPS2*(EPS2*XKNP-EPS1*UKN)
         +EPS1*EPS1*ZKNP
      RKNP=EPS4*(EPS4*XKNP-EPS3*UKN)
         +EPS3*EPS3*ZKNP
      GO TO 230
  220 KKNP=ZETA9*VKNP+ZETA7*WKNP-ZETA4
         *(ZETA6*RKNP-ZETA2*SKNP)
      LKNP=ZETA3*(ZETA2*VKNP-ZETA5*WKNP)
         +ZETA7*RKNP+ZETA8*SKNP
      UKNP=-(EPS1*EPS4+EPS2*EPS3)*UKNP
         +EPS2*EPS4*KKNP+EPS1*EPS3*LKNP
      VKNP=EPS15*(ZETA3*(ZETA6*VKNP+ZETA2
         *WKNP)+ZETA9*RKNP-ZETA7*SKNP)
      SKNP=EPS15*(-ZETA7*VKN+ZETA8*WKNP
         +ZETA4*(ZETA2*RKNP+ZETA5*SKNP))
      WKNP=EPS4*(-EPS4*KKNP+EPS3*UKN)
         -EPS3*EPS3*LKNP
      RKNP=EPS2*(-EPS2*KKNP+EPS1*UKN)
         -EPS1*EPS1*LKNP
  230 CONTINUE
  240 RALPHN=-DSQRT(1.0-CSQ/ALPNSQ)
      RBETAN=-DSQRT(1.0-CSQ/BETNSQ)
      RALRBT=RALPHN*RBETAN
      EPSILN=-EPSILO*CSQ**2/RALRBT
      IF((2*(N/2)).EQ.N) GO TO 250
      FRAYL=EPSILN*(-VKNP*RBETAN+WKNP
         -RKNP*RALRBT-SKNP*RALPHN)
      GO TO 260
```

```
250  FRAYL=EPSILN*(VKNP*RALPHN+RKNP
        -WKNP*RALRBT+SKNP*RBETAN)
260  RETURN
     END
C
C    THIS IS SUBPROGRAM FOR COMPUTAION OF
C    COVRIANCE MATRIX AND RESOLUTION MATR-
C    IX

     SUBROUTINE CALSRC(C)
     IMPLICIT DOUBLE PRECISION(A-H, O-Z)
     DIMENSION BB(40 40),C(40,40), R(40,40), DDX(40)
     COMMON/NM/N, M/PL/PLMD/S/SS
     WRITE(*, 200)((C(I, J), J=1, N), I=1, N)
200  FORMAT(10(15X, 10F14.6))
     DY=SQRT(SS/(M-N))
     WRITE(*, 250) DY
250  FORMAT(10X, F14.9)
     DO 320 I=1, N
     DO 320 J=1, N
     IF(I.EQ.J) GO TO 300
     BB(I, J)=C(I, J)
     GO TO 320
300  BB(I, J)=C(I, J)+PLMD
320  CONTINUE
     CALL GJ1(N, BB, ISW)
     IF(ISW.EQ.1) GO TO 330
     GO TO 1000
330  DO 400 I=1, N
     DO 400 J=1, N
```

```
      R(I, J)=0.0
      DO 400 K=1, N
      R(I, J)=BB(I, K)*C(K, J)+R(I, J)
  400 CONTINUE
      WRITE(*, 430)((R(I, J), J=1, N), I=1, N)
  430 FORMAT(10(/5X, 10F14.9))
  440 C=0.0
      DO 450 I=1, N
      DO 450 J=1, N
      DO 450 K=1, N
      C(I, J)=R(I, K)*BB(K, J)+C(I, J)
  450 CONTINUE
      WRITE(*, 460)((C(I, J), J=1, N), I=1, N)
  460 FORMAT(10(/5X, 10F14.6))
      DO 480 I=1, N
  480 DDX(I)=DY*SQRT(C(I, J))
      WRITE(*, 500)(DDX(I), I=1, N)
  500 FORMAT(2(/20X, 10F14.6))
 1000 RETURN
      END
C
C     THIS IS SUBPROGRAM FOR COMPUTATION ON
C     THE
C     INVERSION OF SYMMETIC MATRIX
C
      SUBROUTINE GJ1(N, A, ISW)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      DIMENSION A(40, 40), H(40)
      DO 100 L=1, N
      K=N-L+1
      P=A(1, 1)
      IF(P.GT.0.0) GO  TO 10
```

```
      ISW=0
      RETURN
   10 DO 50 I=2, N
      Q=A(I, 1)
      IF(I.GT.K) GO  TO  20
      H(I)=-Q/P
      GOTO 30
   20 H(I)=Q/P
   30 DO 40 J=2, I
   40 A(I-1), J-1)=A(I, J)+Q*H(J)
   50 CONTINUE
      A(N, N)=1/P
      DO 60 I=2, N
   60 A(N, I-1)=H(I)
  100 CONTINUE
      L=N-1
      DO 300 I=1, L
      K=I+1
      DO 200 J=K, N
  200 A(I, J)=A(J, I)
  300 CONTINUE
      ISW=1
      RETURN
      END
```

# 附录 计算实例

## （一）Love波频散曲线正演结果

下面给出单层地壳模型计算结果的打印格式。

# COMPUTATION OF THE THEORETIC VALUE FOR LOVE WAVE DISPERSION

| THICKNESS | SHEAR WAVE VELOCITY | DENSITY |
|---|---|---|
| 3.50 | 3.50 | 2.70 |
| * * *(半无限空间) | 4.50 | 3.30 |

| PERIOD | PHASE VELOCITY | GROUP VELOCITY |
|---|---|---|
| 5.0 | 3.524424628 | 3.478699140 |
| 6.0 | 3.534461540 | 3.470868456 |
| 7.0 | 3.545980492 | 3.462388082 |
| 8.0 | 3.558892612 | 3.453472330 |
| 9.0 | 3.573111580 | 3.444330912 |
| 10.0 | 3.588552206 | 3.435169886 |
| 11.0 | 3.605129230 | 3.426192042 |
| 12.0 | 3.622756338 | 3.417596750 |
| 13.0 | 3.641345376 | 3.409579174 |
| 14.0 | 3.660805788 | 3.402329140 |
| 15.0 | 3.681044256 | 3.39602937 |
| 16.0 | 3.701964586 | 3.390853322 |
| 17.0 | 3.723467810 | 3.386962702 |
| 18.0 | 3.745452556 | 3.384504622 |
| 19.0 | 3.767815636 | 3.383608670 |
| 20.0 | 3.790452862 | 3.384383950 |
| 22.0 | 3.836134254 | 3.391266300 |
| 24.0 | 3.881685312 | 3.405521274 |
| 26.0 | 3.926354996 | 3.427071210 |
| 28.0 | 3.969491764 | 3.455396874 |
| 30.0 | 4.010573466 | 3.489607496 |
| 32.0 | 4.049222742 | 3.528551042 |
| 34.0 | 4.085206502 | 3.570948222 |
| 37.0 | 4.133989426 | 3.638250080 |

| | | |
|---|---|---|
| 40.0 | 4.176647430 | 3.706664516 |
| 43.0 | 4.213606982 | 3.773302516 |
| 47.0 | 4.255072314 | 3.856242026 |
| 50.0 | 4.281156742 | 3.912804890 |
| 60.0 | 4.344975782 | 4.064935000 |
| 80.0 | 4.411930384 | 4.243485106 |

## （二） Rayleigh波频散曲线正演结果

**1．大陆型单层地壳模型Rayleigh波频散曲线的计算数据及打印格式**

COMPUTATON OF THE THEORETIC VALUE
FOR RAYLEIGH WAVE DISPERSION

| THICKNESS | SHEAR WAVE VELOCITY | COMPRESSIONAL WAVE VELOCITY | DENSITY |
|---|---|---|---|
| 38. | 3.55 | 6.14 | 2.74 |
| * * *（无限厚层） | 4.65 | 8.20 | 3.32 |

| PERIOD | PHASE VELOCITY | GROUP VELOCITY |
|---|---|---|
| 5.0 | 3.263276538 | 3.262959550 |
| 6.0 | 3.263436524 | 3.261639130 |
| 7.0 | 3.263987176 | 3.257925932 |
| 8.0 | 3.265315966 | 3.250456712 |
| 9.0 | 3.267870804 | 3.238278966 |
| 10.0 | 3.272093322 | 3.220971756 |
| 11.0 | 3.278379778 | 3.198579778 |
| 12.0 | 3.287065388 | 3.171506322 |
| 13.0 | 3.298420932 | 3.140427562 |
| 14.0 | 3.312653230 | 3.106239000 |
| 15.0 | 3.329904678 | 3.070026620 |
| 16.0 | 3.350249736 | 3.033049050 |
| 17.0 | 3.373687742 | 2.996726542 |

| | | |
|---|---|---|
| 18.0 | 3.400132676 | 2.962615988 |
| 19.0 | 3.426401568 | 2.932370888 |
| 20.0 | 3.461204502 | 2.907672042 |
| 22.0 | 3.530702332 | 2.881122610 |
| 24.0 | 3.604285056 | 2.892378422 |
| 26.0 | 3.676880744 | 2.943165410 |
| 28.0 | 3.744082800 | 3.026136322 |
| 30.0 | 3.803223516 | 3.128210294 |
| 32.0 | 3.853505370 | 3.236120870 |
| 34.0 | 3.895428736 | 3.340173750 |
| 37.0 | 3.945115444 | 3.478223816 |
| 40.0 | 3.982594236 | 3.590805682 |
| 43.0 | 4.011274732 | 3.680111702 |
| 47.0 | 4.040033460 | 3.770452180 |
| 50.0 | 4.056636596 | 3.821948330 |
| 60.0 | 4.094047668 | 3.931510650 |
| 80.0 | 4.132983960 | 4.025088822 |

2. 海洋型单层地壳模型 Rayleigh 波频散曲线计算值及打印格式

COMPUTATION OF THE THEORETIC VALUE FOR RAYLEIGH WAVE DISPERSION IN OCEANIC AREA

| THICKNESS | SHEAR WAVE VELOCITY | COMPRESSIONAL WAVE VELOCITY | DENSITY |
|---|---|---|---|
| 6.0 | | 1.52 | 1.03 |
| 6.0 | 3.98 | 6.96 | 2.75 |
| * * *（无限厚层） | 4.68 | 8.10 | 3.09 |

| PERIOD | PHASE VELOCITY | GROUP VELOCITY |
|---|---|---|
| 7.0 | 1.665065234 | 1.370104788 |
| 10.0 | 1.892038910 | 1.204591834 |
| 11.0 | 2.016165512 | 1.134915906 |

| | | |
|---|---|---|
| 12.0 | 2.181902976 | 1.059982386 |
| 13.0 | 2.409301952 | 0.986568570 |
| 14.0 | 2.726273556 | 0.940305268 |
| 15.0 | 3.138857040 | 1.019058140 |
| 16.0 | 3.530854190 | 1.440051150 |
| 17.0 | 3.770657506 | 2.129142146 |
| 18.0 | 3.898293444 | 2.717734700 |
| 19.0 | 3.972063486 | 3.112796994 |
| 20.0 | 4.019399686 | 3.368390800 |
| 21.0 | 4.052376106 | 3.538447534 |
| 22.0 | 4.076795752 | 3.656463736 |
| 23.0 | 4.095726726 | 3.741809626 |
| 24.0 | 4.110926994 | 3.805819316 |
| 25.0 | 4.123471746 | 3.855351368 |
| 26.0 | 4.134054376 | 3.894713910 |
| 27.0 | 4.143141982 | 3.926712562 |
| 28.0 | 4.151060874 | 3.953234834 |
| 29.0 | 4.158046116 | 3.975588502 |
| 30.0 | 4.164271552 | 3.994703300 |
| 32.0 | 4.174939052 | 4.025747382 |
| 36.0 | 4.191327026 | 4.069526196 |
| 40.0 | 4.203513266 | 4.099378468 |
| 43.0 | 4.210857220 | 4.116417196 |
| 45.0 | 4.215120166 | 4.126021314 |
| 50.0 | 4.224105566 | 4.145665656 |
| 60.0 | 4.237250796 | 4.173219544 |
| 80.0 | 4.253383574 | 4.205736280 |

## （三）反 演 实 例

利用1971年8月23日发生在千岛群岛附近一次6.2级地震(45.6°N, 151.0°E)。该地震通过蒙古高原，并由甘肃省高台地

震台记录到。本实例用该台面波测定的 Rayleigh 波群速度作数值反演。选择四层模型进行计算。约束条件取为：地壳总厚度 $H$ 在40—50公里范围内变化，地壳内横波速度的平均值$\overline{V_S}$ 限 制为3.4—3.7公里/秒。

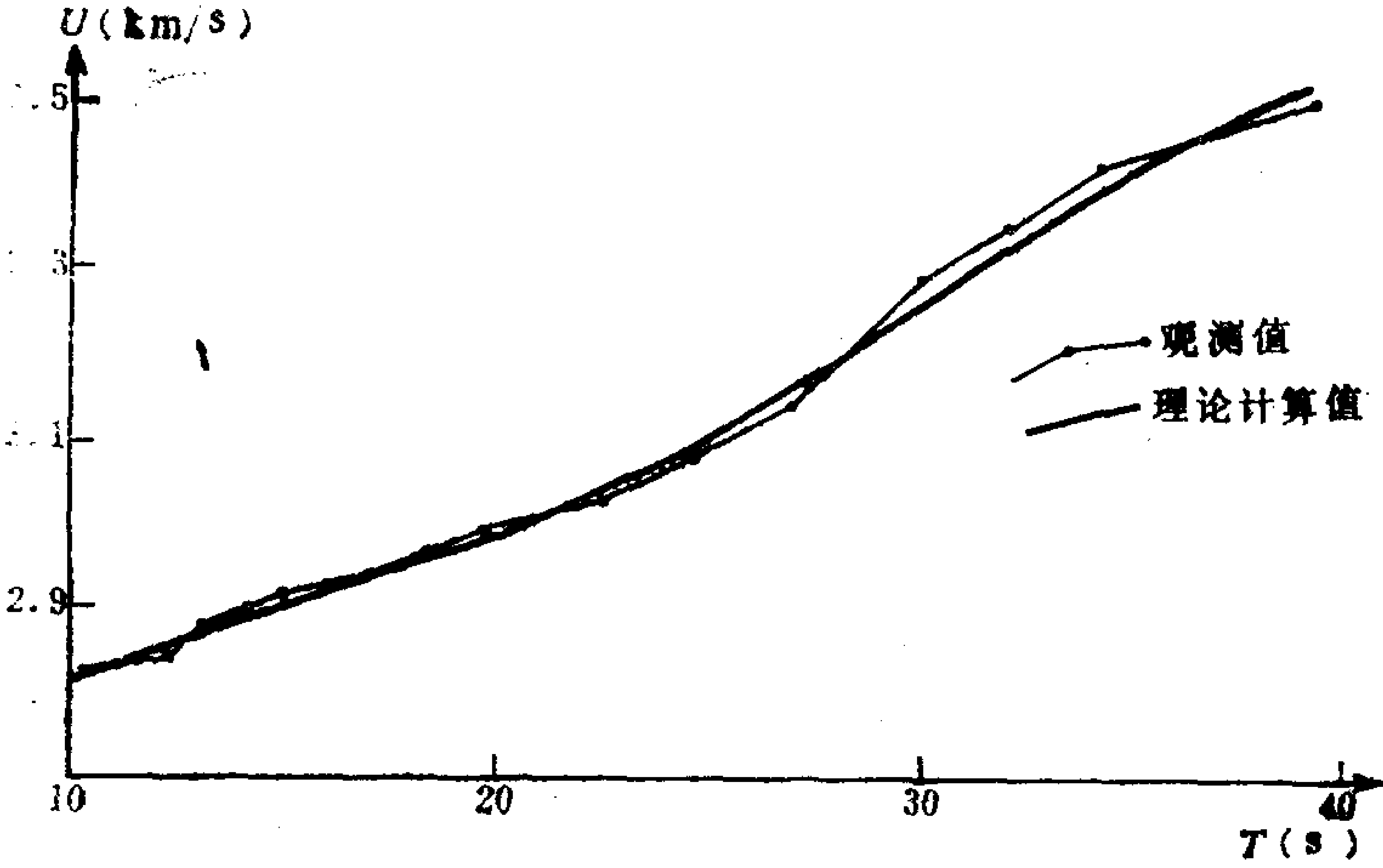
实测频散值与理论计算值如下表所示：

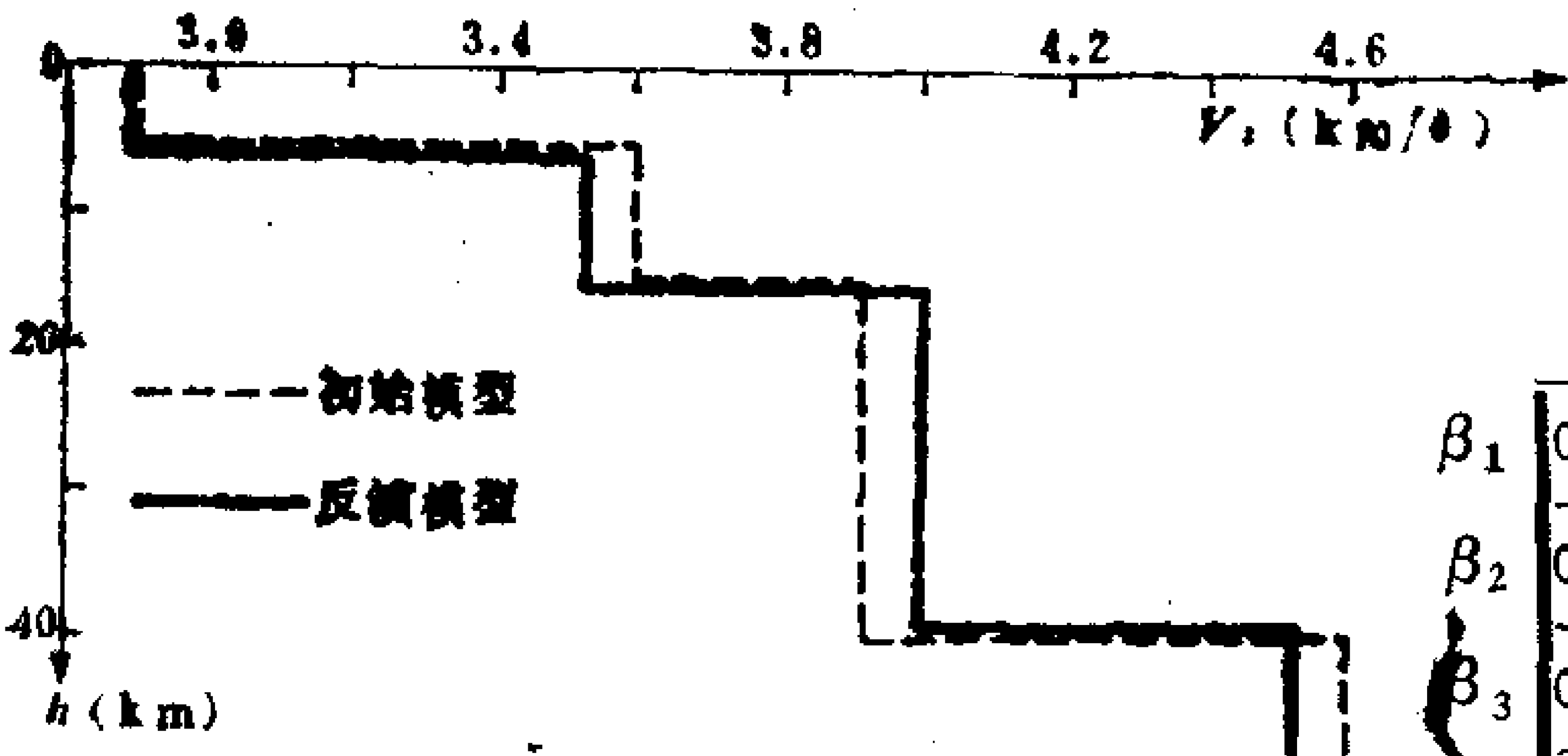
| $T$(s) | $U_{实测}$ (km/s) | $U_{理论}$ (km/s) | T(s) | $U_{实测}$ (km/s) | $U_{理论}$ (km/s) |
|---|---|---|---|---|---|
| 10.0 | 2.820 | 2.811 | 19.7 | 2.990 | 2.985 |
| 11.1 | 2.830 | 2.831 | 22.3 | 3.030 | 3.040 |
| 12.2 | 2.840 | 2.849 | 24.4 | 3.080 | 3.094 |
| 13.1 | 2.860 | 2.865 | 26.9 | 3.140 | 3.168 |
| 14.2 | 2.880 | 2.885 | 28.4 | 3.210 | 3.215 |
| 15.1 | 2.910 | 2.901 | 30.1 | 3.290 | 3.269 |
| 16.0 | 2.930 | 2.917 | 32.0 | 3.350 | 3.329 |
| 17.1 | 2.940 | 2.937 | 34.1 | 3.420 | 3.392 |
| 18.3 | 2.960 | 2.958 | 39.4 | 3.500 | 3.943 |
| 19.0 | 2.980 | 2.971 | | | |

反演结果如图 8-附 3-1 示。由上述结果可见。蒙古高原地壳速度分层清晰，可以用沉积岩、花岗岩和玄武岩典型大陆地壳模型表示。其康腊界面的深度约为22km。莫霍界面深度约 45km。数值反演结果与该区构造特征是一致的。

初始模型及反演结果如下示：

| 层次 | 初始值 | | 反演值 | |
|---|---|---|---|---|
| | 厚度(km) | β(km/s) | 厚度(km) | β(km/s) |
| 1 | 6.0 | 2.90 | 5.8 | 2.90 |
| 2 | 15.0 | 3.60 | 14.6 | 3.52 |
| 3 | 25.0 | 3.90 | 24.8 | 3.99 |
| 4 | ∞ | 4.60 | ∞ | 4.51 |




| | | | | |
|---|---|---|---|---|
| $\beta_1$ | 0.7228 | 0.1242 | 0.0338 | 0.0833 |
| $\beta_2$ | 0.1242 | 0.9367 | −0.0191 | −0.0644 |
| $\beta_3$ | 0.0338 | −0.0191 | 0.9818 | −0.0498 |
| $\beta_4$ | 0.0833 | −0.0649 | −0.0498 | 0.7493 |

图 8-附3-1

# 第九章　理论地震图的计算

利用地震波研究地球内部结构，是广泛开展的一项工作。过去人们利用天然地震和人工爆破，借助于走时和视速度对地壳-上地幔做过一些解释，但结果很粗糙。对震相的振幅只是进行了试探性的研究。为了进一步利用地震资料，深入开展对地球内部结构的研究，一些地震学家又提出了理论地震图方法。理论地震图是根据对地球内部介质分布及震源的某些假设，由基波迭加而成的理论波形图，又称为合成地震图。它的主要用途是利用地震波动力学特征，对观测记录进行对比解释。理论地震图可以有效地帮助解释地震波的振幅变化和波形特征。

当前计算理论地震图的方法一般分成两大类：一类是在时间域里进行；另一类是在频率域内进行。在时间域内主要是利用数字方法，例如差分法和有限单元法。由于在时间域内进行计算难度较大，目前这类方法开展得并不广泛。在频率域内主要是求带有适当辐射及边界条件的解，其次是计算反变换（Fourier 反变换或 Leplace 反变换）。这类方法主要有：(1) 射线理论法，其中包括精确射线理论（Müller, 1968）、广义射线理论（Gilbert, 1972; Wiggins, 1974）、全波理论（Comier, 1977）、射线展开理论〔Kennet, 1974, 1975〕等；(2)反射率法〔Fuchs, 1968〕；(3)渐近近似计算（Chapman, 1978）。

目前理论地震图的研究已由均匀介质进入到非均匀介质的范围。特别是对于横向非均匀介质的研究已开始突破并取得新的进展。

本章将主要介绍计算理论地震图的反射率法。这种方法在目前研究地壳-上地幔结构时得到了广泛的应用。它的特点精度高，

能提供精细的模型。特别有利于用人工爆破资料对地壳构造进行研究。对于计算理论地震图的另一种方法——广义射线理论法，只介绍一般原理。

## §9-1 用反射率法计算理论地震图的基本原理

用反射率法计算理论地震图是一种频谱方法，也就是先要在频率域内求解波动方程。根据适当的辐射及边界条件得到频率域内的地动位移，然后再做反变换（Fourier反变换或Laplace反变换）得到时间域内的地动位移。下面我们详细推导其计算公式。

### （一）基本关系式

地球内部是不均匀物质组成的。为方便起见，我们假定地壳是由分层均匀，各向同性的完全弹性平面层状介质组成。选取如图9-1-1所示的柱对称坐标系，爆炸点位于原点，接收点$S$在地表，即图中的$r$轴上，$\alpha_i$，$\beta_i$分别表示第$i$层的纵波速度和横波速度，它们与介质弹性常数的关系为

$$\alpha_i=\sqrt{\frac{\lambda_i+2\mu_i}{\rho_i}},$$

$$\beta_i=\sqrt{\frac{\mu_i}{\rho_i}}.$$

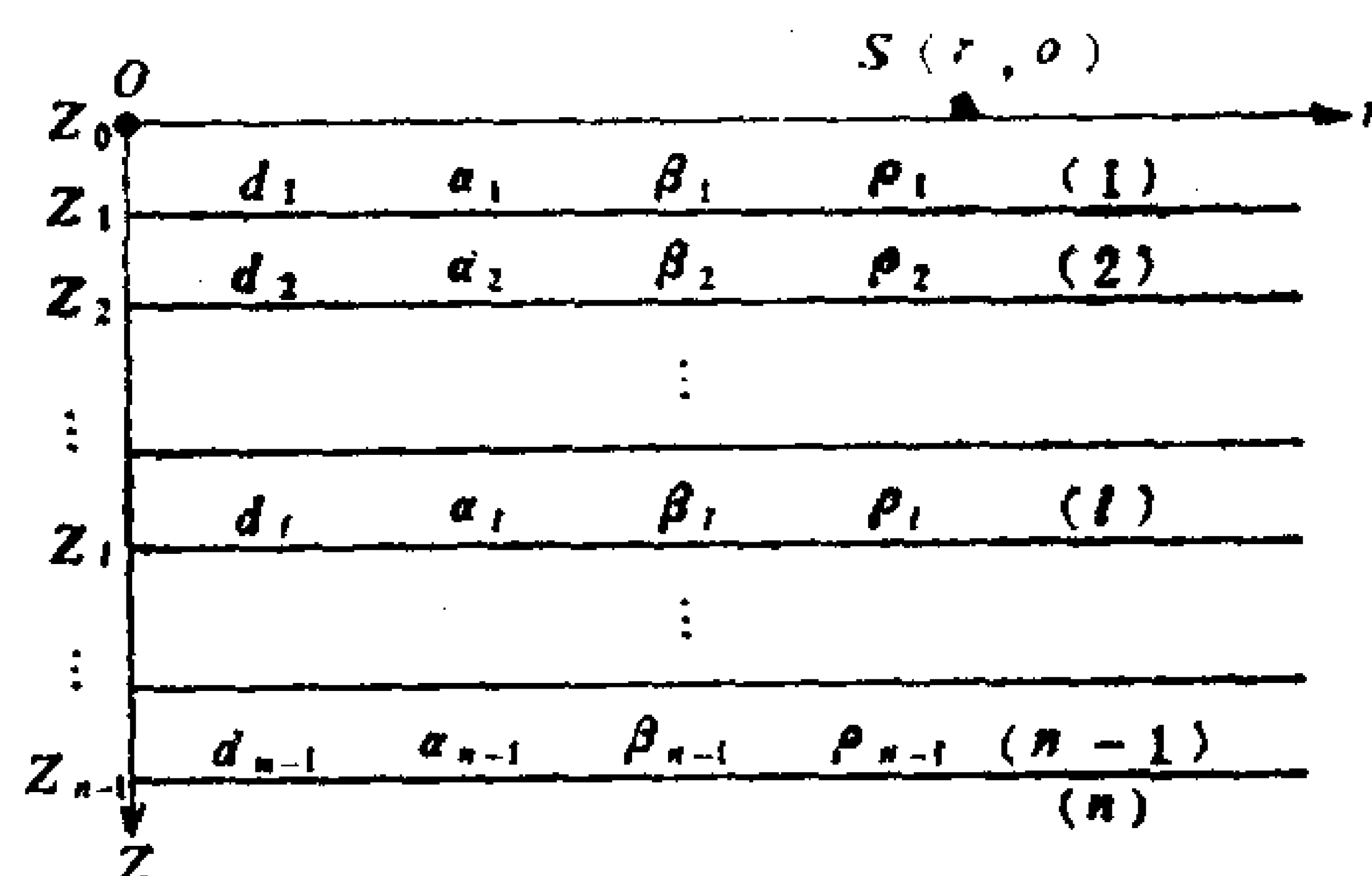



图9-1-1 带有点源爆破的层状介质模型

这里下角标 $i$ 表示层的序号，$\lambda_i$ 为第 i 层的 Lame 常数，$\mu_i$ 为第 $i$ 层的切变模量，$\rho_i$ 为第 $i$ 层的密度。
以

$$d_i = z_i - z_{i-1}$$

表示第 $i$ 层的厚度，其中 $z_i$ 为 第 $i$ 个 界面的埋藏深度。在不涉及到特定层的时候，我们将略去表示层的角标符号。

在接收点的水平位移 $q$ 和垂直位移 $w$ 由纵波 位 $\varphi$ 和 横波位 $\psi$ 得到：

$$q = \frac{\partial\varphi}{\partial r} - \frac{\partial\psi}{\partial z}, \tag{9-1.1}$$

$$w = \frac{\partial\varphi}{\partial z} + \frac{\partial\psi}{\partial r} + \frac{\psi}{r}. \tag{9-1.2}$$

这里 $\varphi(r, z, t)$ 和 $\psi(r, z, t)$ 分别满足方程

$$\nabla^2\varphi = \frac{1}{a^2}\frac{\partial^2\varphi}{\partial t^2}, \tag{9-1.3}$$

$$\nabla^2\psi - \frac{\psi}{r^2} = \frac{1}{\beta^2}\frac{\partial^2\psi}{\partial t^2}. \tag{9-1.4}$$

利用 Fourier 变换式

$$\left.\begin{aligned} \overline{\varphi}(r,z,\omega) &= \int_{-\infty}^{\infty}\varphi(r,z,t)e^{-j\omega t}dt \\ \varphi(r,z,t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\overline{\varphi}(r,z,\omega)e^{j\omega t}d\omega \end{aligned}\right\} \tag{9-1.5}$$

$$\left.\begin{aligned} \overline{\psi}(r,z,\omega) &= \int_{-\infty}^{\infty}\psi(r,z,t)e^{-j\omega t}dt \\ \psi(r,z,t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\overline{\psi}(r,z,\omega)e^{j\omega t}d\omega \end{aligned}\right\} \tag{9-1.6}$$

我们可以得到 $\overline{\varphi}$ 和 $\overline{\psi}$ 所满足的 Fourier 变换方程

$$\nabla^2\overline{\varphi} = -k_a^2\overline{\varphi}, \tag{9-1.7}$$

$$\nabla^2 \bar{\psi} - \frac{\bar{\psi}}{r^2} = -k_\beta^2 \bar{\psi} .\qquad (9\text{-}1.8)$$

这里 $k_\alpha = \frac{\omega}{\alpha}$， $k_\beta = \frac{\omega}{\beta}$，

其中 $\omega$ 表示圆频率。

爆炸点源产生的纵波位，可以写成下式：

$$\varphi_1(r,z,t) = \frac{1}{R} F\left(t - \frac{R}{\alpha_1}\right), \qquad (9\text{-}1.9)$$

这里 $R$ 表示射线离开震源的距离。

上式的 Fourier 变换式为：

$$\overline{\varphi_1}(r,z,\omega) = \overline{F}(\omega) \int_0^\infty J_0(kr) e^{-j\nu_1 z} \frac{kdk}{j\nu_1}. \qquad (9\text{-}1.10)$$

这里 $\overline{F}(\omega)$ 表示只依赖于时间的震源函数的 Fourier 变换，$k = \frac{\omega}{c}$ 为水平波数，$c$ 为相速度，$\nu_1 = \sqrt{k_{\alpha_1}^2 - k^2}$ 为垂直波数，$J_0(kr)$ 为第一类零阶 Bessel 函数。

假定第 $i$ 层的纵波位和横波位是

$$\overline{\varphi}_i(r,z,\omega) = \overline{F}(\omega) \int_0^\infty A_i e^{j\nu_i (z - z_{i-1})} J_0(kr) \frac{kdk}{j\nu_1} + \overline{F}(\omega) \int_0^\infty B_i e^{-j\nu_i (z - z_{i-1})} J_0(kr) \frac{kdk}{j\nu_1}, \qquad (9\text{-}1.11)$$

$$\overline{\psi}_i(r,z,\omega) = \overline{F}(\omega) \int_0^\infty C_i e^{j\nu_i' (z - z_{i-1})} J_1(kr) \frac{k}{\nu_1} dk + \overline{F}(\omega) \int_0^\infty D_i e^{-j\nu_i' (z - z_{i-1})} J_1(kr) \frac{k}{\nu_1} dk. \qquad (9\text{-}1.12)$$

在(9-1.11)和(9-1.12)式中

$$\nu_i = \begin{cases} \sqrt{k_{\alpha_i}^2 - k^2} & c \geqslant \alpha_i \\ -j\sqrt{k^2 - k_{\alpha_i}^2} & c < \alpha_i \end{cases},$$

$$\nu_i' = \begin{cases} \sqrt{k_{\beta i}^2 - k^2} & c \geq \beta_i \\ -j\sqrt{k^2 - k_{\beta i}^2} & c < \beta_i \end{cases}.$$

符号 $A_i$，$B_i$，$C_i$，$D_i$ 表示积分常数， 它们都是$K$的函数， 并依赖于介质参数$\alpha_i$，$\beta_i$，$\rho_i$ 和 $d_i$；$J_1(kr)$ 为第一类一阶 Bessel 函数。在(9-1·11)和(9-1·12)式中，第一项表示沿着 $z$ 轴负向传播的波，第二项表示沿着 $z$ 轴正向传播的波。

取

$$\left.\begin{aligned} \Phi_i^- &= A_i e^{j\nu_i(z-z_{i-1})}, \\ \Phi_i^+ &= B_i e^{-j\nu_i(z-z_{i-1})}, \\ \Psi_i^- &= C_i e^{j\nu_i'(z-z_{i-1})}, \\ \Psi_i^+ &= D_i e^{-j\nu_i'(z-z_{i-1})}. \end{aligned}\right\} \qquad (9\text{-}1\cdot13)$$

把(9-1.13)式代入(9-1.11)和(9-1.12)式，则有：

$$\overline{\varphi}_i(r,z,\omega) = \overline{F}(\omega)\int_0^{\infty}(\Phi_i^- + \Phi_i^+)J_0(kr)\frac{k}{j\nu_1}dk, \quad (9\text{-}1\cdot14)$$

$$\overline{\Psi}_i(r,z,\omega) = \overline{F}(\omega)\int_0^{\infty}(\overline{\Psi}_i^- + \overline{\Psi}_i^+)J_1(kr)\frac{k}{\nu_1}dk. \quad (9\text{-}1\cdot15)$$

## （二）边 界 条 件

求解波动方程(9-1.3)和(9-1.4)必须首先知道问题的边界条件。在界面埋深为 $z_i$ 的第 $i$ 个界面上，位移 $q_i$ 和 $q_{i+1}$，$w_i$ 和 $w_{i+1}$，应力分量 $(\sigma_{zz})_i$ 和 $(\sigma_{zz})_{i+1}$，$(\sigma_{zr})_i$ 和 $(\sigma_{zr})_{i+1}$ 处处连续，这些条件对于 Fourier 变换量同样成立，于是有

$$\left.\begin{aligned} \overline{q}_i &= \overline{q}_{i+1}, \\ \overline{w}_i &= \overline{w}_{i+1}, \\ (\overline{\sigma}_{zz})_i &= (\overline{\sigma}_{zz})_{i+1}, \\ (\overline{\sigma}_{zr})_i &= (\overline{\sigma}_{zr})_{i+1}. \end{aligned}\right\} \qquad (9\text{-}1\cdot16)$$

在上式中$\overline{\sigma}_{zz}$和$\overline{\sigma}_{zr}$分别表示正应力$\sigma_{zz}$和切应力$\sigma_{zr}$的Fourier 变换，其表达式是

$$(\overline{\sigma}_{zz})_i=\lambda_i\nabla^2\overline{\varphi}_i+2\mu_i\left(\frac{\partial^2\overline{\varphi}_i}{\partial z^2}+\frac{\partial^2\overline{\psi}_i}{\partial r\partial z}+\frac{1}{r}\frac{\partial\overline{\psi}_i}{\partial z}\right),\tag{9-1·17}$$

$$(\overline{\sigma}_{zr})_i=\mu_i\left(2\frac{\partial^2\overline{\varphi}_i}{\partial r\partial z}-\frac{\partial^2\overline{\psi}_i}{\partial z^2}+\frac{\partial^2\overline{\psi}_i}{\partial r^2}+\frac{1}{r}\frac{\partial\overline{\psi}_i}{\partial r}-\frac{\overline{\psi}_i}{r^2}\right).\tag{9-1.18}$$

在自由表面上正应力和切应力为零，相应的Fourier 变换表达式为

$$\left.\begin{aligned}(\overline{\sigma}_{zz})_0=0,\\(\overline{\sigma}_{zr})_0=0.\end{aligned}\right\}\tag{9-1·19}$$

## （三）反 射 系 数

用反射率法计算理论地震图的核心是求界面反射系数（即反射率）。本节我们将详细叙述反射系数的导出过程，同时给出可避免有效数字损失的计算表达式。

### 1．反射系数的基本表达式

对(9-1.1)和(9-1.2)式两边取 Fourier 变换得到（此处带上角标）

$$\left.\begin{aligned}\overline{q}_i&=\frac{\partial\overline{\varphi}_i}{\partial r}-\frac{\partial\overline{\psi}_i}{\partial z},\\\overline{w}_i&=\frac{\partial\overline{\varphi}_i}{\partial z}+\frac{\partial\overline{\psi}_i}{\partial r}+\frac{\overline{\psi}_i}{r}.\end{aligned}\right\}\tag{9-1·20}$$

把(9-1.14)和(9-1.15)式代入(9-1.20)式得到

$$\overline{q}_i=\overline{F}(\omega)\int_0(-k)(\overline{\Phi}_i^-+\Phi_i)J_1(kr)\frac{kdk}{j\nu_1}$$

$$-F(\omega)\int_0(j\nu_i'\Psi_i^--j\nu_i'\Psi_i)J_1(kr)\frac{kdk}{\nu_1}$$

$$= \overline{F}(\omega)\int_0^{\infty}\overline{Q}_i J_1(kr)\frac{kdk}{\nu_1} . \qquad (9\text{-}1\cdot21)$$

其中

$$\overline{Q}_i = jk\Phi_i^- - j\nu_i'\Psi_i^- + jk\Phi_i^+ + j\nu_i'\Psi_i^+ . \qquad (9\text{-}1\cdot22)$$

用同样的方法可以求得

$$\overline{w}_i = \overline{F}(\omega)\int_0^{\infty}\overline{W}_i J_0(kr)\frac{kdk}{j\nu_1} , \qquad (9\text{-}1\cdot23)$$

$$(\overline{\sigma}_{zz})_i = \overline{F}(\omega)\int_0^{\infty}(\overline{P}_{zz})_i J_0(kr)\frac{kdk}{j\nu_1} , \qquad (9\text{-}1\cdot24)$$

$$(\overline{\sigma}_{zr})_i = \overline{F}(\omega)\int_0^{\infty}(\overline{P}_{zr})_i J_1(kr)\frac{kdk}{\nu_1} . \qquad (9\text{-}1\cdot25)$$

其中

$$\overline{W}_i = j\nu_i\Phi_i^- + jk\Psi_i^- - j\nu_i\Phi_i^+ + jk\Psi_i^+ , \qquad (9\text{-}1\cdot26)$$

$$(\overline{P}_{zz})_i = \mu_i l_i\Phi_i^- - 2\mu_i k\nu_i'\Psi_i^- + \mu_i l_i\Phi_i^+ + 2\mu_i k\nu_i'\Psi_i^+ , \qquad (9\text{-}1\cdot27)$$

$$(\overline{P}_{zr})_i = -2\mu_i k\nu_i\Phi_i^- - \mu_i l_i\Psi_i^- + 2\mu_i k\nu_i\Phi_i^+ - \mu_i l_i\Psi_i^+ . \qquad (9\text{-}1\cdot28)$$

这里 $l_i = 2k^2 - k_{\beta i}^2$。

从(9-1.21)至(9-1.28)式我们可以看出采用矩阵法求解是非常方便的。下面定义一个位矢量 $\boldsymbol{\Phi}_i$ 和一个位移-应力矢量 $\boldsymbol{S}_i$：

$$\boldsymbol{\Phi}_i = (\Phi_i, \Psi_i^-, \Phi_i^+, \Psi_i^+)^T , \qquad (9\text{-}1\cdot29)$$

$$\boldsymbol{S}_i = (\overline{Q}_i, \overline{W}_i, (\overline{P}_{zz})_i, (\overline{P}_{zr})_i)^T . \qquad (9\text{-}1\cdot30)$$

这里 $T$ 表示矩阵转置。

（9-1.29）与（9-1.30）式之间有如下关系

$$S_i = T_i \Phi_i . \qquad (9\text{-}1\cdot 31)$$

上式中的矩阵 $T_i$ 为系数矩阵，由层参数决定，它可以直接从〔9-1.22〕，（9-1.26），〔9-1.27〕和（9-1.28）式的联合得到，其表达式如下：

$$T_i = \begin{pmatrix} +jk & -j\nu_i' & +jk & +j\nu_i' \\ +j\nu_i & +jk & -j\nu_i & +jk \\ +\mu_i l_i & -2\mu_i k\nu_i' & +\mu_i l_i & +2\mu_i k\nu_i' \\ -2\mu_i k\nu_i & -\mu_i l_i & +2\mu_i k\nu_i & -\mu_i l_i \end{pmatrix}. \qquad (9\text{-}1\cdot 32)$$

从（9-1.31）式中解出 $\Phi_i$，

$$\Phi_i = T_i^{-1} S_i \ , \qquad (9\text{-}1\cdot 33)$$

其中 $T_i^{-1}$ 为矩阵 $T_i$ 的逆矩阵，其表达式是：

$$T_i^{-1} = \frac{\beta_i^2}{2\mu_i \omega^2 \nu_i \nu_i'} \begin{pmatrix} -j2\mu_i k\nu_i\nu_i' & +j\mu_i l_i \nu_i' & -\nu_i\nu_i' & -k\nu_i' \\ -j\mu_i l_i \nu_i & -j2\nu_i k\nu_i\nu_i' & -k\nu_i & +\nu_i\nu_i' \\ -j2\mu_i k\nu_i\nu_i' & -j\mu_i l_i \nu_i' & -\nu_i\nu_i' & +k\nu_i' \\ +j\mu_i l_i \nu_i & -j2\mu_i k\nu_i\nu_i' & +k\nu_i & +\nu_i\nu_i' \end{pmatrix}. \qquad (9\text{-}1\cdot 34)$$

$T_i^{-1}$ 的正确性可以通过下式来验证：

$$T_i T_i^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

（9-1.31）式与（9-1.33）式对于第 $i$ 层内的每一深度都是正确的，并且在 $i$ 层顶部的 $\Phi_i$ 与在 $i$ 层底部的 $\Phi_i$ 之间有如下关系：

$$\Phi_i(z_i) = E_i \Phi_i(z_{i-1}) . \qquad (9\text{-}1\cdot 35)$$

上式中

$$
\boldsymbol{E}_i = \begin{pmatrix} e^{j\nu_i d_i} & & & 0 \\ & e^{j\nu_i d_i} & & \\ & & e^{-j\nu_i d_i} & \\ 0 & & & e^{-j\nu_i d_i} \end{pmatrix}. \tag{9-1·36}
$$

对任何距离 $r$ 都必须满足 $n-1$ 个界面上的应力及位移的连续条件，根据(9-1-16)式我们有

$$
\boldsymbol{S}_i(z_i)=\boldsymbol{S}_{i+1}(z_i). \tag{9-1·37}
$$

在无穷远处的辐射条件由(9-1.11)和(9-1.12)式中指数的符号选择来满足。在下面半空间没有入射能量。所以有

$$
\left.\begin{aligned} \Phi_n^- &= 0, \\ \Psi_n^- &= 0. \end{aligned}\right\} \tag{9-1·38}
$$

在第一层中只有 $P$ 波入射到反射层，所以有

$$
\Psi_1^+ = 0. \tag{9-1·39}
$$

比较(9-1.10)与(9-1.11)式，我们得到

$$
\Phi_1^+(z)=e^{-j\nu_1 z}, \tag{9-1·40}
$$

当 $z=z_1$ 时，有

$$
\boldsymbol{\Phi}_1(z_1)=(A_1 e^{-j\nu_1 d_1},\ C_1 e^{j\nu'_1 d_1},\ e^{-j\nu_1 d_1},\ 0)^T. \tag{9-1·41}
$$

上式中的 $d_1=z_1-z_0=d_1$.

当 $z=z_{n-1}$ 时，有

$$
\boldsymbol{\Phi}_n(z_{n-1})=(0,\ 0,\ B_n,\ D_n)^T. \tag{9-1·42}
$$

令

$$
A_1=\tilde{R}_{PP}e^{-j2d_1\nu_1}, \tag{9-1·43}
$$

$$C_1 = \tilde{R}_{PS}\ e^{-jd_1(\nu_1-\nu_1')}, \tag{9-1·44}$$

$$B_n = \tilde{T}_{PP}\ e^{-jd_1\nu_1}, \tag{9-1·45}$$

$$D_n = \tilde{T}_{PS}\ e^{-jd_1\nu_1}. \tag{9-1·46}$$

这里$\tilde{R}_{PP}$和$\tilde{T}_{PP}$分别表示 $P$ 波到 $P$ 波的反射系数和透射系数，$\tilde{R}_{PS}$和$\tilde{T}_{PS}$分别表示 $P$ 波到 $S$ 波的反射系数和透射系数。这样矢量 $\boldsymbol{\Phi}_1^T$ 和 $\boldsymbol{\Phi}_n^T$可以写成

$$\boldsymbol{\Phi}_1(z_1) = e^{-j\nu_1 d_1}(\tilde{R}_{PP},\ \tilde{R}_{PS},\ 1,\ 0)^T, \tag{9-1·47}$$

$$\boldsymbol{\Phi}_n(z_{n-1}) = e^{-j\nu_1 d_1}(0,\ 0,\ \tilde{T}_{PP},\ \tilde{T}_{PS})^T. \tag{9-1·48}$$

由(9-1·31)式知

$$\begin{aligned} &\boldsymbol{S}_1(z_1) = \boldsymbol{T}_1\boldsymbol{\Phi}_1(z_1), \\ &\boldsymbol{S}_2(z_1) = \boldsymbol{S}_1(z_1) = \boldsymbol{T}_1\boldsymbol{\Phi}_1(z_1), \end{aligned} \tag{9-1·49}$$

而

$$\boldsymbol{S}_2(z_1) = \boldsymbol{T}_2\boldsymbol{\Phi}_2(z_1). \tag{9-1·50}$$

所以

$$\begin{aligned} &\boldsymbol{\Phi}_2(z_1) = \boldsymbol{T}_2^{-1}\boldsymbol{S}_2(z_1), \\ &\boldsymbol{\Phi}_2(z_2) = \boldsymbol{E}_2\boldsymbol{\Phi}_2(z_1) = \boldsymbol{E}_2\boldsymbol{T}_2^{-1}\boldsymbol{T}_1\boldsymbol{\Phi}_1(z_1), \\ &\boldsymbol{S}_2(z_2) = \boldsymbol{T}_2\boldsymbol{E}_2\boldsymbol{T}_2^{-1}\boldsymbol{T}_1\boldsymbol{\Phi}_1(z_1). \end{aligned} \tag{9-1·51}$$

令

$$\boldsymbol{G}_2 = \boldsymbol{T}_2\boldsymbol{E}_2\boldsymbol{T}_2^{-1}, \tag{9-1·52}$$

则

$$\boldsymbol{S}_2(z_2) = \boldsymbol{G}_2\boldsymbol{T}_1\boldsymbol{\Phi}_1(z_1). \tag{9-1·53}$$

依此类推可以得到下列递推公式

$$\boldsymbol{S}_i(z_i) = \boldsymbol{G}_i\boldsymbol{G}_{i-1}\cdots\boldsymbol{G}_3\cdot\boldsymbol{G}_2\cdot\boldsymbol{T}_1\boldsymbol{G}_1(z_1). \tag{9-1·54}$$

其中 $\boldsymbol{G}_i$ 为关于另一条对角线对称的 $4\times4$ 矩阵，其元素表达式如下：

$$(G_i)_{11}=(G_i)_{44}=-\gamma_i\cos P_i+(\gamma_i+1)\cos Q_i,$$

$$(G_i)_{12}=(G_i)_{34}=j[(\gamma_i+1)r_{\alpha_i}^{-1}\ \sin P_i+\gamma_i r_{\beta_i}\sin Q_i],$$

$$(G_i)_{13}=(G_i)_{24}=j(\rho_i\cdot c\cdot\omega)^{-1}(\cos Q_i-\cos P_i),$$

$$(G_i)_{21}=(G_i)_{43}=-j[\gamma_i r_i\sin P\gamma+(\gamma_i+1)r_{\beta_i}^{-1}\sin Q_i],$$

$$(G_i)_{14}=(\rho_i\cdot c\cdot\omega)^{-1}(r_{\alpha_i}^{-1}\sin P_i+r_{\beta_i}\sin Q_i),$$

$$(G_i)_{22}=(G_i)_{33}=(\gamma_i+1)\cos P_i-\gamma_i\cos Q_i,$$

$$(G_i)_{33}=-(\rho_i\cdot c\cdot\omega)^{-1}(r_{\alpha_i}\sin P_i-r_{\beta_i}^{-1}\sin Q_i)$$

$$(Q_i)_{31}=(G_i)_{42}=-j\rho_i\cdot c\cdot\omega\cdot\gamma_i(\nu_i+1)(\cos P_i-\cos Q_i),$$

$$(G_i)_{32}=-\rho_i\cdot c\cdot\omega[(\gamma_i+1)^2 r_{\alpha_i}^{-1}\sin P_i+\nu_i^2 r_{\beta_i}\sin Q_i],$$

$$(G_i)_{41}=-\rho_i\cdot c\cdot\omega[\gamma_i^2 r_{\alpha_i}^{-1}\sin P_i+(\gamma_i+1)^2 r_{\beta_i}^{-1}\sin Q_i].$$

(9-1·55)

在上式中

$$P_i=k\cdot r_{\alpha_i}d_i,$$

$$Q_i=k\cdot r_{\beta_i}d_i,$$

$$r_i=-2\beta_i^2/c^2,$$

$$r_{\alpha_i}=\begin{cases}[(c/\alpha_i)^2-1]^{1/2} & c\geqslant\alpha_i\\ -j[1-(c/\alpha_i)^2]^{1/2} & c<\alpha_i\end{cases},$$

$$r_{\beta_i}=\begin{cases}[(c/\beta_i)^2-1]^{1/2} & c\geqslant\beta_i\\ _i-j[1-(c/\beta_i)^2]^{1>2} & c<\alpha_i\end{cases}.$$

在第 $n-1$ 个界面上有

$$S_n(z_{n-1})=S_{n-1}(z_{n-1})=G_{n-1}\cdot G_{n-2}\cdots G_2\cdot T_1\Phi_1(z_1),$$

(9-1·56)

$$\Phi_n(z_{n-1})=T_n^{-1}\ S_n(z_{n-1})=T_n^{-1}G_{n-1}G_{n-2}\cdots G_2T_1\Phi_1(z_1).$$

(9-1·57)

令

$$M=T_n^{-1}G_{n-1}G_{n-2}\cdots G_2\cdot T_1, \qquad (9\text{-}1\cdot58)$$

于是(9-1.57)式变成

$$\Phi_n(z_{n-1})=M\Phi_1(z_1). \qquad (9\text{-}1\cdot59)$$

很显然 $M$ 仍为$4\times4$矩阵。把(9-1·47)和(9-1·48)式代入(9-1·59)式，我们得到

$$e^{-j\nu_1 d_1}(0,\ 0,\ \widetilde{T}_{PP},\ \widetilde{T}_{PS})^T=e^{-j\nu_1 d_1}M(\widetilde{R}_{PP},\ \widetilde{R}_{PS},1,0)^T. \qquad (9\text{-}1\cdot60)$$

消去指数项，则有

$$(0,\ 0,\ \widetilde{T}_{PP},\ \widetilde{T}_{PS})^T=M(\widetilde{R}_{PP},\ \widetilde{R}_{PS},\ 1,\ 0)^T. \qquad (9\text{-}1\cdot61)$$

把$M$分块取

$$M=\begin{bmatrix}M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix}, \qquad (9\text{-}1\cdot62)$$

这里$M_{ij}$为 $2\times2$ 子矩阵。把(9-1·62)式代入(9-1·61)式并展开得到

$$\begin{pmatrix}0\\0\end{pmatrix}=M_{11}\begin{pmatrix}\widetilde{R}_{PP}\\ \widetilde{R}_{PS}\end{pmatrix}+M_{12}\begin{pmatrix}1\\0\end{pmatrix},$$

$$\begin{pmatrix}\widetilde{T}_{PP}\\ \widetilde{T}_{PS}\end{pmatrix}=M_{21}\begin{pmatrix}\widetilde{R}_{PP}\\ \widetilde{R}_{PS}\end{pmatrix}+M_{22}\begin{pmatrix}1\\0\end{pmatrix}.$$

即

$$\begin{pmatrix}\widetilde{R}_{PP}\\ \widetilde{R}_{PS}\end{pmatrix}=-M_{11}^{-1}M_{12}\begin{pmatrix}1\\0\end{pmatrix}, \qquad (9\text{-}1\cdot63)$$

$$\begin{pmatrix} \tilde{T}_{PP} \\ \tilde{T}_{PS} \end{pmatrix} = (-M_{21}M_{11}^{-1}M_{12} + M_{22})\begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{9-1·64}$$

设$M_{11}$的行列式为 $\det M_{11}$，伴随矩阵为$\tilde{M}_{11}$，于是

$$M_{11}^{-1} = \tilde{M}_{11}/\det M_{11}. \tag{9-1·65}$$

以 $m_{ij}$ 表示矩阵 $M$ 的元素，即可得到$\tilde{M}_{11}$ 的矩阵形式，

$$\tilde{M}_{11} = \begin{bmatrix} m_{22} & -m_{12} \\ -m_{21} & m_{11} \end{bmatrix}, \tag{9-1·66}$$

$$\det M_{11} = m_{11}m_{22} - m_{12}m_{21}. \tag{9-1·67}$$

把(9-1.66)和(9-1.67)式代入(9-1.63)式得到

$$\begin{pmatrix} \tilde{R}_{PP} \\ \tilde{R}_{PS} \end{pmatrix} = -\frac{1}{\det M_{11}}\begin{pmatrix} m_{22}m_{13} & -m_{12}m_{23} \\ m_{11}m_{23} & -m_{21}m_{13} \end{pmatrix}. \tag{9-1·68}$$

把上式写开则有

$$\tilde{R}_{PP} = (m_{12}m_{23} - m_{13}m_{22})/\det M_{11}, \tag{9-1·69}$$

$$\tilde{R}_{PS} = -(m_{11}m_{23} - m_{13}m_{21})/\det M_{11}. \tag{9-1·70}$$

把(9-1.69)和(9-1.70)式代入(9-1.64)式中得到

$$\tilde{T}_{PP} = m_{31}\tilde{R}_{PP} + m_{32}\tilde{R}_{PS} + m_{33}, \tag{9-1·71}$$

$$\tilde{T}_{PS} = m_{41}\tilde{R}_{PP} + m_{22}\tilde{R}_{PS} + m_{34}. \tag{9-1·72}$$

利用 (9-1·69)，(9-1·70)，(9-1·71) 和(9-1·72)式即可计算反射系数$\tilde{R}_{PP}$ 和$\tilde{R}_{PS}$ 以及透射系数 $\tilde{T}_{PP}$和 $\tilde{T}_{PS}$ 。但这样计算存在严重的缺点，因为直接利用$M$ 矩阵的元素计算反射系数会出现有效数字的损失，因此(9-1·69)和(9-1·70)式不能直接运用，必须加以改进。下面将讨论改进方法。

**2. 改进后的反射系数表达式**

为了避免上述问题，这一节讨论改进措施。改进将按照下面的运算规则进行：

$$m\left|\begin{matrix}ij\\kl\end{matrix}\right. = m_{ik}m_{jl} - m_{il}m_{jk}. \tag{9-1·73}$$

在上式中，$i$，$j$ 表示矩阵$M$的第 $i$，$j$ 行；$k$，$l$ 表示矩阵的第 $k$，$l$ 列。根据(9-1.37)式，我们可以由矩阵$M$构造出 $\delta$ 矩阵 $\widehat{M}$，由矩阵 $T_n^{-1}$ 构造出 $\delta$ 矩阵 $\widetilde{T}_n$，由矩阵 $T_1$ 构造出 $\delta$ 矩阵$\widehat{T}_1$ 由矩阵 $G_i$ 构造出 $\delta$ 矩阵 $\widehat{G}_i$ 。为了书写方便，我们把 $\widehat{T}_n$, $T_1$ 及 $\widehat{G}_i$ 的下角标改写成上角标并用小括号括起。按照(9-1·73)式可以求出矩阵$\widehat{T}^{(n)}$ 的部分元素表达式：

$$\widehat{T}_{11}^{(n)} = T_n^{-1}\left\|\begin{matrix}12\\12\end{matrix}\right. = (T_n^{-1})_{11}(T_n^{-1})_{22} - (T_n^{-1})_{12}(T_n^{-1})_{21}$$

$$= \frac{-\beta_n^4\rho_n}{2\omega^2}(4k^2\nu_n\nu_n' + l_n^2),$$

$$\widehat{T}_{12}^{(n)} = T_n^{-1}\left|\begin{matrix}12\\13\end{matrix}\right. = j\nu_n/2,$$

$$\widehat{T}_{13}^{(n)} = T_n^{-1}\left|\begin{matrix}12\\14\end{matrix}\right. = -j\frac{\beta_n^2}{2\omega c}(l_n + 2\nu_n\nu_n'),$$

$$\widehat{T}_{14}^{(n)} = T_n^{-1}\left\|\begin{matrix}12\\23\end{matrix}\right. = \widetilde{T}_{13}^{(n)},$$

$$\widehat{T}_{15}^{(n)} = T_n^{-1}\left\|\begin{matrix}12\\24\end{matrix}\right. = -j\nu_n'/2,$$

$$\widehat{T}_{16}^{(n)} = T_n^{-1}\left\|\begin{matrix}12\\34\end{matrix}\right. = \frac{1}{2\rho_n\omega^2}(\nu_n\nu_n' + k^2), \tag{9-1·75}$$

矩阵$\widehat{G}^{(i)}$ 的元素表达式：

$$\widehat{G}_{11}^{(i)} = G_i\left|\begin{matrix}12\\12\end{matrix}\right. = \widetilde{G}_{66}^{(i)} = G_i\left\|\begin{matrix}34\\34\end{matrix}\right. = -2\gamma_i(\gamma_i+1) + [2\gamma_i(\gamma_i$$

$$+1)+1]\cdot\cos P_i\cdot\cos Q_i - [(\Upsilon_i+1)^2 U_i Y_i$$

$$+\gamma_i^2 X_i Z_i],$$

$$\widehat{G}_{12}^{(i)} = G_i \Big\|_{13}^{12} = \widehat{G}_{56}^{(i)} = G_i \Big\|_{34}^{24} = (\rho_i \cdot c \cdot \omega)^{-1}(Y_i \cos P_i$$

$$+ X_i \cos Q_i),$$

$$\widehat{G}_{13}^{(i)} = G_i \Big\|_{14}^{12} = \widehat{G}_{14}^{(i)} = G_i \Big\|_{23}^{12} = \widehat{G}_{36}^{(i)} = G_i \Big\|_{34}^{14} = \widehat{G}_{46}^{(i)}$$

$$= G_i \Big\|_{34}^{23} = j(\rho_i \cdot c \cdot \omega)^{-1}[(2\gamma_i + 1)(1 - \cos P_i \cos Q_i)$$

$$+ (\gamma_i + 1) \cdot U_i Y_i + \gamma_i X_i Z_i],$$

$$\widehat{G}_{15}^{(i)} = G_i \Big\|_{24}^{12} = \widehat{G}_{26}^{(i)} = G_i \Big\|_{34}^{13} = -(\rho_i \cdot c \cdot \omega)^{-1}(U_i \cos Q_i$$

$$+ Z_i \cos P_i),$$

$$\widehat{G}_{16}^{(i)} = G_i \Big\|_{34}^{12} = -(\rho_i \cdot c \cdot \omega)^{-2}[2(1 - \cos P_i \cdot \cos Q_i)$$

$$+ U_i Y_i + X_i Z_i],$$

$$\widehat{G}_{21}^{(i)} = G_i \Big\|_{12}^{13} = \widehat{G}_{65}^{(i)} = G_i \Big\|_{24}^{34}$$

$$= -(\rho_i \cdot c \cdot \omega)[\gamma_i^2 \cdot Z_i \cdot \cos P_i + (\gamma_i + 1)^2 \cdot U_i$$

$$\times \cos Q_i],$$

$$\widehat{G}_{22}^{(i)} = G_i \Big\|_{13}^{13} = \widehat{G}_{55}^{(i)} = G_i \Big\|_{24}^{24} = \cos P_i \cdot \cos Q_i,$$

$$\widehat{G}_{23}^{(i)} = G_i \Big\|_{14}^{13} = \widehat{G}_{24}^{(i)} = G_i \Big\|_{23}^{13} = \widehat{G}_{35}^{(i)} = G_i \Big\|_{24}^{14} = \widehat{G}_{45}^{(i)}$$

$$= G_i \Big\|_{24}^{23} = j[(\gamma_i + 1) U_i \cos Q_i + \gamma_i Z_i \cos P_i],$$

$$\widehat{G}_{25}^{(i)} = G_i \Big\|_{24}^{13} = U_i \cdot Z_i,$$

$$\widehat{G}_{31}^{(i)} = G_i \Big\|_{12}^{14} = \widehat{G}_{41}^{(i)} = G_i \Big\|_{12}^{23} = \widehat{G}_{63}^{(i)} = G_i \Big\|_{14}^{34} = \widetilde{G}_{64}^{(i)}$$

$$= G_i \Big\|_{23}^{34} = j(\rho_i \cdot c \cdot \omega)\{\gamma_i(\gamma_i + 1)(2\gamma_i + 1)(1 -$$

$$-\cos P_i \cdot \cos \cdot Q_i) + [(\Upsilon_i + 1)^3 U_i \cdot Y_i + \Upsilon_i^3 \cdot X_i \cdot Z_i]\},$$

$$\widehat{G}_{32}^{(i)} = G_i \left|\right|_{13}^{14} = \widehat{G}_{42}^{(i)} = G_i \left|\right|_{13}^{23} = \widehat{G}_{53}^{(i)} = G_i \left|\right|_{14}^{24} = \widehat{G}_{54}^{(i)}$$

$$= G_i \left|\right|_{23}^{24} = -j[(\Upsilon_i + 1) Y_i \cos P_i + \Upsilon_i \cdot X_i \cdot \cos Q_i],$$

$$\widehat{G}_{33}^{(i)} = G_i \left|\right|_{14}^{14} = \widehat{G}_{44}^{(i)} = G_i \left|\right|_{23}^{23} = 1 + 2\Upsilon_i(\Upsilon_i + 1)(1$$

$$-\cos P_i \cos Q_i) + (\Upsilon_i + 1)^2 U_i Y_i + \Upsilon_i^2 X_i Z_i,$$

$$\widehat{G}_{34}^{(i)} = G_i \left|\right|_{23}^{14} = \widehat{G}_{43}^{(i)} = G_i \left|\right|_{14}^{23} = \widehat{G}_{33}^{(i)} - 1,$$

$$\widehat{G}_{51}^{(i)} = G_i \left|\right|_{12}^{24} = \widehat{G}_{62}^{(i)} = G_i \left|\right|_{13}^{34} = (\rho_i \cdot c \cdot \omega)[(\Upsilon_i$$

$$+ 1)^2 \cdot Y_i \cos P_i + \Upsilon_i^2 \cdot X_i \cdot \cos Q_i],$$

$$\widehat{G}_{52}^{(i)} = X_i \cdot Y_i,$$

$$\widehat{G}_{61}^{(i)} = -(\rho_i \cdot c \cdot \omega)^2 [2\Upsilon_i^2 (\Upsilon_i + 1)^2 (1 - \cos P_i \cos Q_i)$$

$$+ (\Upsilon_i + 1)^4 \cdot U_i \cdot Y_i + \Upsilon_i^4 \cdot X_i \cdot Z_i],$$

(9-1·75

其中

$$U_i = \Upsilon_{\alpha_i}^{-1} \sin P_i, \qquad Y_i = \Upsilon_{\beta_i}^{-1} \cdot \sin Q_i,$$

$$X_i = \Upsilon_{\alpha_i} \sin P_i, \qquad Z_i = \Upsilon_{\beta_i} \cdot \sin Q_i. \qquad (9\text{-}1\cdot 76)$$

矩阵 $\widehat{T}^{(1)}$ 的部分元素表达式：

$$\widehat{T}_{11}^{(1)} = T_1 \left|\right|_{12}^{12} = -k^2 - \nu_1 \nu_1',$$

$$\widehat{T}_{21}^{(1)} = T_1 \left|\right|_{12}^{13} = -j\rho_1 \nu_1' \omega^2,$$

$$\widehat{T}_{31}^{(1)} = T_1 \left|\right|_{12}^{14} = \widehat{T}_{41}^{(1)} = T_1 \left|\right|_{12}^{23} = -j\rho_1 \beta_1^2 k\ (l_1 + 2\nu_1 \nu_1'),$$

$$\widehat{T}_{51}^{(1)} = T_1 \left|\right|_{12}^{24} = \widehat{T}_{54}^{(1)} = T_1 \left|\right|_{23}^{24} = j\rho_1 \nu_1 \omega^2,$$

$$\hat{T}_{61}^{(1)} = T_1 \left\|\begin{matrix}34\\12\end{matrix}\right. = -\rho_1^2\beta_1^4(l_1^2 + 4k^2\nu_1\nu_1'),$$

$$\hat{T}_{14}^{(1)} = T_1 \left\|\begin{matrix}12\\23\end{matrix}\right. = -\hat{T}_{13}^{(1)} = -T_1 \left\|\begin{matrix}12\\14\end{matrix}\right. = k^2 - \nu_1\nu_1',$$

$$\hat{T}_{24}^{(1)} = T_1 \left\|\begin{matrix}13\\23\end{matrix}\right. = -\hat{T}_{21}^{(1)} = -T_1 \left\|\begin{matrix}13\\12\end{matrix}\right. = j\rho_1\nu_1'\omega^2,$$

$$\hat{T}_{34}^{(1)} = T_1 \left\|\begin{matrix}14\\23\end{matrix}\right. = \hat{T}_{44}^{(1)} = T_1 \left\|\begin{matrix}23\\23\end{matrix}\right. = -\hat{T}_{33}^{(1)} = -T_1 \left\|\begin{matrix}14\\14\end{matrix}\right.$$

$$= j\rho_1\beta_1^2 k(l_1 - 2\nu_1\nu_1'),$$

$$\hat{T}_{64}^{(1)} = T_1 \left\|\begin{matrix}34\\23\end{matrix}\right. = -\hat{T}_{63}^{(1)} = -T_1 \left\|\begin{matrix}34\\14\end{matrix}\right. = \rho_1^2 \cdot \beta_1^4(l_1^2$$

$$- 4k^2\nu_1\nu_1'),$$

$$\hat{T}_{12}^{(1)} = T_1 \left\|\begin{matrix}12\\13\end{matrix}\right. = 2k\nu_1,$$

$$\hat{T}_{22}^{(1)} = T_1 \left\|\begin{matrix}13\\13\end{matrix}\right. = \hat{T}_{52}^{(1)} = T_1 \left\|\begin{matrix}24\\13\end{matrix}\right. = 0,$$

$$\hat{T}_{32}^{(1)} = T_1 \left\|\begin{matrix}14\\13\end{matrix}\right. = j4\rho_1\beta_2^1 k^2\nu_1,$$

$$\hat{T}_{42}^{(1)} = T_1 \left\|\begin{matrix}23\\13\end{matrix}\right. = j2\rho_1\beta_1^2 l_1\nu_1,$$

$$\hat{T}_{62}^{(1)} = T_1 \left\|\begin{matrix}34\\13\end{matrix}\right. = 4\rho_1^2\beta_1^1 l_1\nu_1 k,$$

$$\hat{T}_{15}^{(1)} = T_1 \left\|\begin{matrix}12\\24\end{matrix}\right. = 2k\nu_1',$$

$$\hat{T}_{35}^{(1)} = T_1 \left\|\begin{matrix}14\\24\end{matrix}\right. = j2u_1 l_1\nu_1',$$

$$\hat{T}_{45}^{(1)} = T_1 \left\|\begin{matrix}23\\24\end{matrix}\right. = j4\rho_1\beta_1^2 k^2\nu_1',$$

$$\hat{T}_{53}^{(1)} = T_1 \left\|\begin{matrix}24\\14\end{matrix}\right. = j\rho_1\beta_1^2\nu_1(2k^2 - l_1). \tag{9-1·77}$$

矩阵$\hat{M}$可以由$M$得到。但由于$\delta$矩阵有一个特殊而重要的性质，即矩阵乘积的$\delta$矩阵等于每个$\delta$矩阵的乘积，因而$\hat{M}$矩阵又

可直接由$\hat{T}^{(n)}$、$\hat{G}^{(i)}$和$\hat{T}^{(1)}$的乘积得到，即

$$\hat{M}=\hat{T}^{(n)}\cdot\hat{G}^{(n-1)}\cdots\hat{G}^{(2)}\cdot\hat{T}^{(1)}. \qquad (9\text{-}1\cdot78)$$

计算反射系数$\tilde{R}_{PP}$和$\tilde{R}_{PS}$可以直接利用矩阵$\hat{M}$的元素。按照(9-1.73)式的原则由(9-1.69)和(9-1.70)式得到

$$\tilde{R}_{PP}=\frac{m_{12}m_{23}-m_{13}m_{22}}{m_{11}m_{22}-m_{12}m_{21}}=\left(m\left\|\begin{matrix}12\\23\end{matrix}\right.\right)\Big/\left(m\left\|\begin{matrix}12\\12\end{matrix}\right.\right)=\hat{m}_{14}/\hat{m}_{11}, \qquad (9\text{-}1\cdot79)$$

$$\tilde{R}_{PS}=-\frac{m_{11}m_{23}-m_{13}m_{21}}{m_{11}m_{22}-m_{12}m_{21}}=\left(-m\left\|\begin{matrix}12\\13\end{matrix}\right.\right)\Big/\left(m\left\|\begin{matrix}12\\12\end{matrix}\right.\right)$$

$$=-\hat{m}_{12}/\hat{m}_{11}. \qquad (9\text{-}1\cdot80)$$

在上面的两个表达式中，$\hat{m}_{11}$，$\hat{m}_{12}$和$\hat{m}_{14}$表示矩阵$\hat{M}$的元素。由此我们可以看出只要计算出$\hat{M}$矩阵的某些元素就可算出反射系数的值。实际上我们只需要做下面形式的矩阵乘法。用符号矩阵表示为：

$$(1\times6)_n(6\times6)_{n-1}\cdots(6\times6)_i\cdots(6\times6)_2(6\times3)_1. \qquad (9\text{-}1\cdot81)$$

这里$(1\times6)_n$表示矩阵$\hat{T}^{(n)}$的第一行，$(6\times6)_i$表示矩阵$\hat{G}^{(i)}$，$(6\times3)_1$表示矩阵$\hat{T}^{(1)}$的第一，二，三列。

从(9-1.75)式可以看出矩阵$\hat{G}^{(i)}$是关于另一条对角线对称的矩阵，并且它的第三，四行或第三，四列元素相同。根据(9-175)式的性质和$\hat{T}^{(n)}_{13}=\hat{T}^{(n)}_{14}$及$\hat{T}^{(1)}_{31}=\hat{T}^{(1)}_{41}$这一特点，我们可以把(1×6)，(6×6)和(6×3)的形式分别写成(1×5)，(5×5)及(5×3)的形式。这样以来(9-1.81)式就变成下面的形式：

$$(1\times5)_n(5\times5)_{n-1}\cdots(5\times5)_i\cdots(5\times5)_2(5\times3)_1 .$$

(9-1·82)

上式的计算量远远少于(9-1·81)式的计算量。这样就缩短了计算时间，提高了计算效率。矩阵$\widehat{G}$可以写成下面形式：

$$\widehat{G}=\begin{pmatrix}\widehat{G}_{11} & \widehat{G}_{12} & 2\widehat{G}_{13} & \widehat{G}_{15} & \widehat{G}_{16}\\ \widehat{G}_{21} & \widehat{G}_{22} & 2\widehat{G}_{23} & \widehat{G}_{25} & \widehat{G}_{26}\\ \widehat{G}_{31} & \widehat{G}_{32} & 2\widehat{G}_{33}-1 & \widehat{G}_{35} & \widehat{G}_{36}\\ \widehat{G}_{51} & \widehat{G}_{52} & 2\widehat{G}_{53} & \widehat{G}_{55} & \widehat{G}_{56}\\ \widehat{G}_{61} & \widehat{G}_{62} & 2\widehat{G}_{63} & \widehat{G}_{65} & \widehat{G}_{66}\end{pmatrix} . \quad (9\text{-}1\cdot83)$$

下面我们来导出$\widehat{R}_{PP}$的计算表达式。在具体导出之前先引用一个定理：

如果$P$，$A_1$，$A_2$，…，$A_n$是具有两个或更多符号排列的矩阵且

$$P=A_1\cdot A_2\cdots A_n ,$$

则

$$p\left|\right|_{kl}^{ij}=a'\left|\right|_{bc}^{ij}\cdot a^2\left|\right|_{ef}^{bc}\cdots a^{n-1}\left|\right|_{uv}^{st}\cdot a^n\left|\right|_{kl}^{uv} . \quad (9\text{-}1\cdot84)$$

在上式中$p$，$a'$，$a^2$，…，$a^n$分别表示矩阵$P$，$A_1$，$A_2$，…，$A_n$的元素。(9-1·84)式中字母符号的迭加对是只有不同字符的不同对。我们可以选取$bc$为12，13，14，23，24，34等，即$b<c$，绝对不能同时包括12和21两者。该定理的证明见Dukin的文章（Dukin，1965）。利用上述定理则可得到

$$\widehat{m}_{11}=m\left|\right|_{12}^{12}=T_n^{-1}\left|\right|_{ab}^{12}\cdot G_{n-1}\left|\right|_{cd}^{ab}\cdots G_2\left|\right|_{uv}^{st}\cdot T_1\left|\right|_{12}^{uv}$$

$$=\widehat{T}_{1i}^{(n)}\cdot\widehat{G}_{ij}\cdot\widehat{T}_{j1}^{(1)} . \quad (9\text{-}1\cdot85)$$

其中

$$\hat{G}_{ij}=(\hat{G}_{n-1})_{ik}\cdot(\hat{G}_{n-2})_{kl}\cdots(\hat{G}_3)_{tu}\cdot(\hat{G}_2)_{uj}.$$

把(9-1.85)式展开并写成矩阵形式则有

$$\hat{m}_{11}=\left(\hat{T}_{11}^{(1)},\ \hat{T}_{21}^{(1)},\ \hat{T}_{31}^{(1)},\ \hat{T}_{51}^{(1)},\ \hat{T}_{61}^{(1)}\right)\cdot\hat{G}^T$$

$$\times\left(\hat{T}_{11}^{(n)},\ \hat{T}_{12}^{(n)},\ 2\hat{T}_{13}^{(n)},\ \hat{T}_{15}^{(n)},\ \hat{T}_{16}^{(n)}\right)^T. \qquad (9\text{-}1\cdot86)$$

上式中$\hat{G}^T$为(9-1.83)式中$\hat{G}$矩阵的转置。利用$\hat{G}$的对称性我们得到

$$\hat{m}_{11}=\hat{T}_{11}^{(1)}\cdot t_1+\hat{T}_{21}^{(1)}\cdot t_2+\hat{T}_{31}^{(1)}\cdot t_3+\hat{T}_{51}^{(1)}\cdot t_4$$

$$+\hat{T}_{61}^{(1)}\cdot t_5. \qquad (9\text{-}1\cdot87)$$

在上式中$t_i(i=1,\ 2,\ \cdots,\ 5)$表示(9-1.86)式中后两个矩阵相乘的乘积元素。根据同样的道理我们有

$$\hat{m}_{14}=-\hat{T}_{13}^{(1)}t_1-\hat{T}_{21}^{(1)}\cdot t_2-\hat{T}_{33}^{(1)}\cdot t_3+\hat{T}_{51}^{(1)}\cdot t_4$$

$$-\hat{T}_{63}^{(1)}t_5. \qquad (9\text{-}1\cdot88)$$

把(9-1·87)和(9-1·88)式代入(9-1·79)式中则有

$$\hat{R}_{PP}=\frac{-\hat{T}_{13}^{(1)}\cdot t_1-\hat{T}_{21}^{(1)}\cdot t_2-\hat{T}_{33}^{(1)}\cdot t_3+\hat{T}_{51}^{(1)}\cdot t_4-\hat{T}_{63}^{(1)}t_5}{\hat{T}_{11}^{(1)}\cdot t_1+\hat{T}_{2}^{(1)}\cdot t_2+\hat{T}_{31}^{(1)}\cdot t_3+\hat{T}_{51}^{(1)}\cdot t_4+\hat{T}_{61}^{(1)}\cdot t_5}. \qquad (9\text{-}1.89)$$

利用上式计算反射系数就可避免有效数字的损失。

## （四）带有自由表面影响的位移

我们的目的是求地震波到达地表面的振动位移，以便与实际观测资料相比较，因此必须考虑自由表面的影响。我们仅考虑$P$波到达地表的位移，

反射带的反射$P$波位可在（9-1.11）式中令$i=1$，取其第一项并代入$A_1$的表达式，于是得到

$$\varphi_r=\overline{F}(\omega)\int_0^{\infty}J_0(kr)\ R_{PP}\cdot e^{j\nu_1(z-2d_1)}\cdot\frac{kdk}{j\nu_1}.$$

（9-1.90）

这个反射$P$波入射到自由表面后又产生两种反射波，其位函数是：

$$\overline{\varphi}_{rr}=\overline{F}(\omega)\int_0^{\infty}J_0(kr)\ \widetilde{R}_{PP}\cdot r_{PP}\cdot e^{-j(z+2d_1)\nu_1}\frac{kdk}{j\nu_1},$$

（9-1.91）

$$\overline{\Psi}_{rr}=\overline{F}(\omega)\int_0^{\infty}J_1(kr)\ \widetilde{R}_{PP}r_{PS}\cdot e^{-j2\nu_1d_1-j\nu_1'd_1}\frac{kdk}{\nu_1}.$$

（9-1.92）

把(9-1.90)，(9-1.91)和(9-1.92)式代入(9-1.20)式可得到

$$\overline{q}(r,\ z,\ \omega)=F(\omega)\int_0^{\infty}\Big[jke^{j\nu_1(z-2d_1)}$$

$$+jKr_{PP}\cdot e^{-j\nu_1(z+2d_1)}$$

$$+j\nu_1'r_{PS}\cdot e^{-j2d_1\nu_1-jz\nu_1'}\Big]\widetilde{R}_{PP}J_0(kr)\frac{kdk}{\nu_1},$$

（9-1.93）

$$\overline{w}(r,\ z,\ \omega)=\overline{F}(\omega)\int_0^{\infty}\Big[\nu_1e^{j\nu_1(z-2d_1)}$$

$$-\nu_1r_{PP}e^{-j\nu_1(z+2d_1)}$$

$$+kr_{PS}\cdot e^{-j2d_1\nu_1-jz\nu_1'}\Big]\widetilde{R}_{PP}J_0(kr)\frac{kdk}{\nu_1}. \tag{9-1.94}$$

在自由表面上 $z=0$，于是(9-1.93)和(9-1.94)式变成：

$$\overline{q}(r,\ 0,\ \omega)=\overline{F}(\omega)\int_0^{\infty}jk\left(1+r_{PP}+r_{PS}\cdot\frac{\nu_1'}{k}\right)J_1(kr)$$

$$\times e^{-j2\nu_1 d_1}\cdot\widetilde{R}_{PP}\frac{k}{\nu_1}dk, \tag{9-1.95}$$

$$\overline{w}(r,\ 0,\ \omega)=\overline{F}(\omega)\int_0^{\infty}\left(1-r_{PP}+\frac{k}{\nu_1}r_{PS}\right)J_0(kr)$$

$$\times e^{-j2\nu_1 d_1}\cdot\widetilde{R}_{PP}kdk. \tag{9-1.96}$$

令

$$\left.\begin{aligned}\Omega&=2k^2-k_{\beta 1}^2,\\ F&=4k^2\nu_1\nu_1'+\Omega^2.\end{aligned}\right\} \tag{9-1.97}$$

于是

$$r_{PP}=(4k^2\nu_1\nu_1'-\Omega^2)/F, \tag{9-1.98}$$

$$r_{PS}=-4k\nu_1\Omega/F. \tag{9-1.99}$$

我们用 $E_q$ 和 $E_w$ 分别表示水平位移和垂直位移的自由表面影响函数，

$$E_q=jk\left(1+r_{PP}+\frac{\nu_1'}{k}r_{PS}\right)=j4k\nu_1\nu_1'k_{\beta 1}^2/F, \tag{9-1.100}$$

$$E_w=1-r_{PP}+\frac{k}{\nu_1}r_{PS}=-2k_{\beta 1}^2\Omega/F. \tag{9-1.101}$$

如果我们所考虑的模型在反射层的上方还有一组覆盖层，如图 9-1-2 所示。图中的O为爆破源，S为接收点，$m$ 为覆盖层的

层数，其余的参数与图9-1-1相同，在覆盖层中只考虑透射情形，相应的 $\bar{q}$ 及 $\bar{w}$ 变成下面的形式：

$$\bar{q}(r,\ o,\ \omega)=F(\omega)\int_0^{\infty}\widetilde{R}_{PP}E_qT_{pd}T_{pu}J_1(kr)\times\exp\left(-2j\sum_{i=1}^{m}d_i\nu_i\right)\frac{k}{\nu_1}dk, \tag{9-1.102}$$

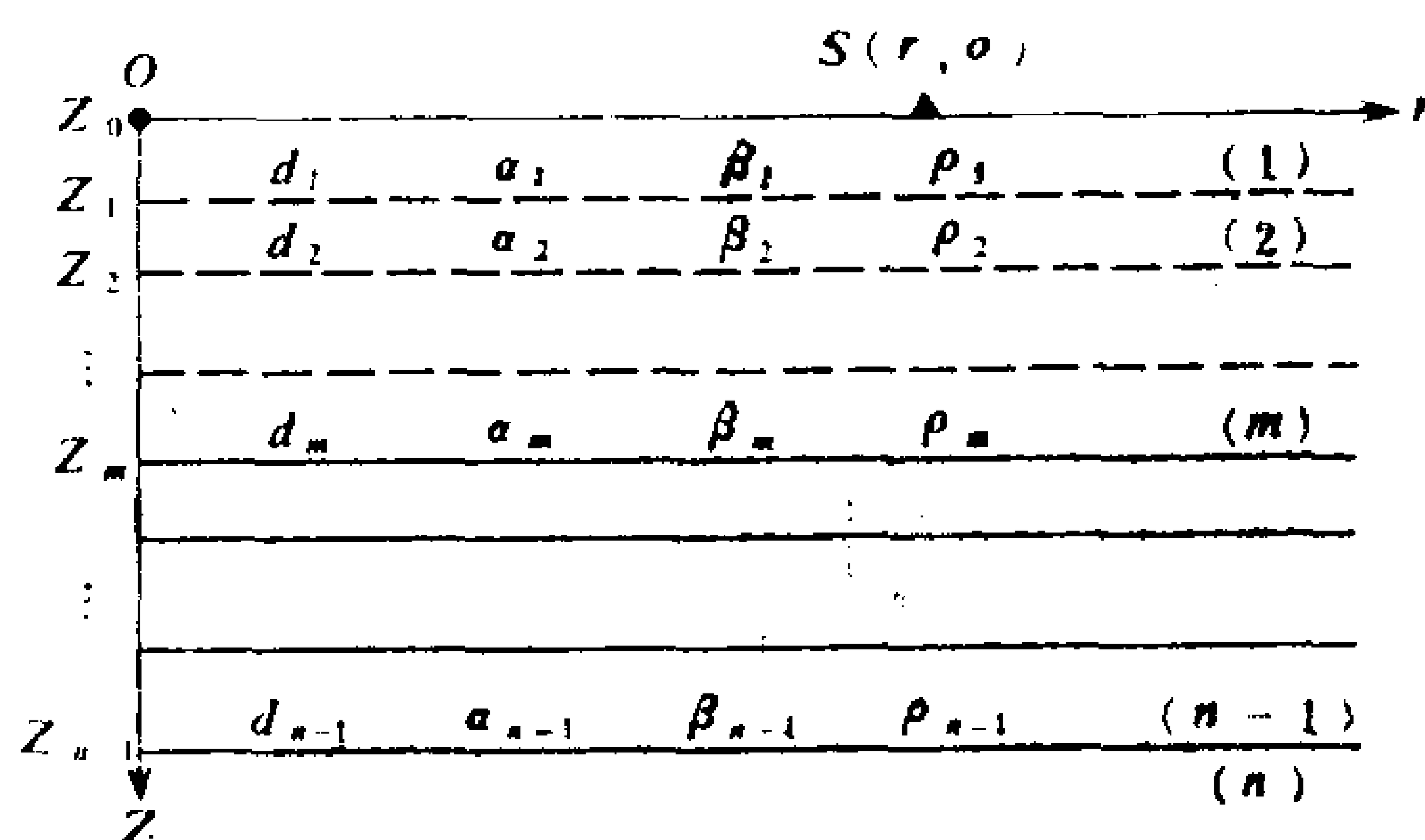



图9-1-2 带有覆盖层的介质模型

$$\bar{w}(r,\ o,\ \omega)=\bar{F}(\omega)\int_0^{\infty}\widetilde{R}_{PP}T_{pd}T_{pu}J_0(kr)E_{\omega}\times\exp\left(-2j\sum_{i=1}^{m}d_i\nu_i\right)kdk. \tag{9-1.103}$$

这里 $T_{pd}$ 是从震源向下传播到反射层的各界面透射系数的乘积，$T_{pu}$ 是从反射层向上传播到第一层的各界面透射系数的乘积；exp 表示底数 e。

由于 $k=\omega/c$，即波数$k$即是相速度 $c$ 的函数又是圆频率$\omega$ 的函数，因此在做数字积分时使用$k$将是不方便的，下面我们引入一个与 $k$ 有关的新变量使其与 $\omega$ 相对独立，命

$$k=\frac{\omega}{\alpha_m}\sin\gamma=k_{\alpha_m}\sin\gamma\ ,\tag{9-1.104}$$

在上式中 $\alpha_m$ 为第 $m$ 层的纵波速度。如果我们把 $\gamma$ 角限制为实数，那么它可以解释为射线在反射层顶部的入射角，引入新变量后，函数 $T_{pd}$，$T_{pu}$ 及 $E_q$ 和 $E_w$ 就只依赖于 $\gamma$ 而与 $\omega$ 无关，积分(9-1.102)和(9-1.103)式变成：

$$\overline{q}(r,\ o,\ \omega)=\overline{\quad}\qquad \frac{2}{\alpha_m}\int_{r_1}^{r_2}\sin\gamma\cos\gamma R_{PP}(\omega,\ \gamma)G(\gamma)$$

$$\times J_1\ (k_{\alpha_m}r\sin\gamma)\exp(-j2k_{\alpha_m}\sum_{i=1}^{m}d_i\eta_i)d\gamma,$$

$$\tag{9-1.105}$$

$$\overline{w}(r,\ o,\ \omega)=\overline{F}(\omega)k_{\alpha_m}^2\int_{r_1}^{\gamma_2}\sin\gamma\cos\gamma J_0\ (k_{\alpha_m}r\sin\gamma)$$

$$\times\widetilde{R}_{PP}(\omega,\gamma)\cdot H(\gamma)\exp(-j2k_{\alpha_m}\sum_{i=1}^{m}d_i\eta_i)d\gamma.$$

$$\tag{9-1.106}$$

函数 $G(\gamma)$ 和 $H(\gamma)$ 的表达式分别是：

$$G(\gamma)=g(\gamma)\prod_{i=2}^{m}T_i(\gamma),\qquad H(\gamma)=h(\gamma)\prod_{i=2}^{m}T_i(\gamma),$$

$$g(\gamma)=4j\left(\frac{\alpha_m}{\beta_1}\right)^2\eta_1'\sin\gamma\ \frac{1}{B},\qquad h(\gamma)=2\left(\frac{\alpha_m}{\beta_1}\right)^2\ \frac{A}{B},$$

$$A=\left(\frac{\alpha_m}{\beta_1}\right)^2-2\sin^2\gamma\ ,\qquad B=A^2+4\eta_1\eta_1'\sin^2\gamma,$$

$$T_i(\gamma)=4\rho_{i-1}\rho_i\eta_{i-1}\eta_i\ \frac{C^2}{D^2}\ ,$$

$$C=\eta_i'(p_i\sin^2\gamma-\rho_{i-1})-\eta_{i-1}'(p_i\sin^2\gamma+\rho_i),$$

$$D=\sin^2\gamma(p_i\sin^2\gamma+\rho_i-\rho_{i-1})^2+\eta_i\eta_i'(p_i\sin^2\gamma-\rho_{i-1})^2$$

$$+\rho_{i-1}\rho_i\eta_i\eta_{i-1}'+\eta_{i-1}[p_i^2\eta_i\eta_i'\eta_{i-1}'\sin^2\gamma+\eta_{i-1}'$$

$$\times(p_i\sin^2\gamma+\rho_i)^2+\rho_{i-1}\rho_i\eta_i'],$$

$$\eta_i=\left[\left(\frac{\alpha_m}{\alpha_i}\right)^2-\sin^2\gamma\right]^{1/2},$$

$$\eta_i'=\left[\left(\frac{\alpha_m}{\beta_i}\right)^2-\sin^2\gamma\right]^{1/2},$$

$$p_i=2\left[\rho_{i-1}\left(\frac{\beta_{i-1}}{\alpha_m}\right)^2-\rho_i\left(\frac{\beta_i}{\alpha_m}\right)^2\right].\qquad(9\text{-}1\cdot107)$$

如果在 $z=0$ 处的位移不受自由表面的影响，即用半空间代替第一层，此时可取 $g(\gamma)=\frac{j\sin\gamma}{\eta_1}$，$h(\gamma)=1$。若 $m=1$，即图 9-1-1 所示的模型情况，则 $G(\gamma)=g(\gamma)$，$H(\gamma)=k(\gamma)$。

由 (9-1.104) 式我们得到：当 $k_1=0$ 时；$\gamma_1=0$，当 $k_2=\infty$ 时，$\gamma_2=\frac{\pi}{2}-j\infty$，这就是(9-1.105) 和 (9-1.106) 式中的积分限，积分路径在 $\gamma=\frac{\pi}{2}$ 处成为一直角。已经证明把积分变量限制为实数且取 $\gamma_1\geqslant0$，$\gamma_2\leqslant\frac{\pi}{2}$ 对于研究体波是足够的。

把(9-1.105)式和(9-1.106)式化成数字积分，利用计算机可得频率域内的位移值 $\overline{q}$ 和 $\overline{w}$。再对 $\overline{q}$ 和 $\overline{w}$ 做 Fourier反变换即可得到时间域内的地动位移 $q$ 和 $w$。把离散的位移值 $q$ 和 $w$ 绘制在时间-距离平面上就可得到理论地震图。

# §9-2 用反射率法计算理论地震图的数字处理方法

## （一） 反射系数的计算

用反射率法计算理论地震图最关键的部分是计算界面反射系数。由反射系数$\widetilde{R}_{PP}$的表达式知，它即是圆频率$\omega$的函数又是角度$\gamma$的函数。换言之，我们要在频率范围（$\omega_1$，$\omega_2$）和角度范围（$\gamma_1$，$\gamma_2$）内计算反射系数。以$\Delta\gamma$表示角度增量，那么可以从下式得到角度样点数：

$$N_\gamma = \frac{\gamma_2 - \gamma_1}{\Delta\gamma} \quad . \qquad (9\text{-}2\cdot1)$$

以$\Delta\omega$表示频率增量，可从下式得到频率样点数：

$$N_\omega = \frac{\omega_2 - \omega_1}{\Delta\omega} \quad . \qquad (9\text{-}2\cdot2)$$

由上面两式可知$\widetilde{R}_{PP}(\omega, \gamma)$的计算总数是$N_\gamma \times N_\omega$，这是一个很大的数，需要占据大量的计算机内存，一般的计算机都很难满足这一条件。

反射率法的一个特点是，反射系数的值对任意接收点距都是一样的，因此只需要计算一次。具体的计算可以采用不同的方法。

第一种方法是先计算反射系数，然后积分。为了避免反射系数占据大量的内存，可使用外部设备。把求得的反射系数值记入外部设备，使用时再从外部设备调入。第二种方法是改变积分顺序。先给定频率值，对$\gamma$角计算反射系数后立即把它们用到不同的接收距，这样就避开了存贮所有反射系数，积分结果需要增加

一个位移数组，但这个数组并不很大。后一种方法不仅可以省去外存，还可以缩短计算时间。

## （二）在计算过程中对参数的控制

用反射率法计算理论地震图精度较高，但计算时间也较长。因此我们希望控制某些参数以便减少计算时间，同时又能保持记录响应的主要特征的精度。

控制方法可分成两类：

*A*.控制频率，持续时间，入射角。

*B*.控制反射带的定义。

*A* 类控制的关键参数是：

①$\Delta t$：理论地震图和数字信号源的时间增量。

②$T$：记录响应的持续时间。

③$(f_1, f_2)$：频率窗。

④$(c_1, c_2)$：相速度窗。

⑤$\Delta Y$：入射角的增量。

*B* 类控制参数包括：

①覆盖层和反射带的定义。

②模拟过渡层。

③地球改平近似。

这里我们将主要讨论 *A* 类控制，对 *B* 类控制只做简单叙述。

### 1. 时间增量

a. 为了避免混淆，时间增量 $\Delta t$ 要选得足够小，以确保在 Nyquist 频率处信号源的振幅谱趋于零。$\Delta t$ 的取值一般可为 0.01或0.02 s等。

b. 通常我们使用一个很小的 $\Delta t$，使 Nyquist 频率大大超过震源的频率范围。

### 2. 记录响应的持续时间

记录响应的持续时间要选得合适，过长会使计算时间冗长，过短则不能包括全部所需要的震相。持续时间一般选取观测记录的最长响应时间。所谓响应时间是指在某一距离范围内观测记录的 $P$ 波震相的实际持续时间，这个时间依赖于震源到接收点的距离。在走时曲线的交叉点附近响应时间很短。而当 $P_g$ 和 $P_n$ 震相在记录图上分开距离很大时，这一时间很长。

3．频率窗

Nyquist 频率是通过时间增量 $\Delta t$ 的选择确定的

$$f_{Ny}=\frac{1}{2\Delta t}. \qquad (9\text{-}2\cdot3)$$

频率增量依赖于响应时间，即

$$\Delta f=\frac{1}{T}. \qquad (9\text{-}2\cdot4)$$

利用(9-2.3)和(9-2.4)式可以确定 Nyquist 频率系数 $Nf_{Ny}$

$$Nf_{Ny}=f_{Ny}/\Delta f=\tfrac{1}{2}\cdot\frac{T}{\Delta t}=\tfrac{1}{2}N_t.$$

上式中的 $N_t$ 表示响应时间样点数。

需要说明的一点是反射系数只需要计算到 Nyquist 频率，在 Nyquist 频率以后的那些值为先前值的共轭复数。

在问题1中已经指出，$\Delta t$ 通常取得很小，因此 Nyquist 频率很大。Nyquist 频率比信号源频率的上限要大得多，但实际计算反射系数时，只需算到信号源的频率上限。如果我们以 $f_1$ 表示信号源的频率下限，以 $f_2$ 表示信号源的频率上限，则频率窗的频带宽为 $f_2-f_1$，频率样点数是

$$N_f=\frac{f_2-f_1}{\Delta f}=T(f_2-f_1). \qquad (9\text{-}2\cdot5)$$

上式表明频率样点数不依赖时间样点数，仅与实际的持续时间有

关。换言之，试图通过增大$\Delta t$来减少时间样点数是不能节约计算时间的。由于$\omega = 2\pi f$，所以(9-2•5)式的$N_f$与(9-2•2)式的$N_\omega$是一样的。

**4. 相速度窗**

在§9-1节中关于波数$k$的积分被变换成关于反射带顶部入射角的积分。每个入射角都相应于一个相速度$c$，

$$c = \alpha_m/\sin\gamma. \tag{9-2•6}$$

其中$\alpha_m$表示与反射带顶部相接的覆盖层中的纵波速度。对于体波，$\gamma$的积分范围是$(0, \pi/2)$，即相速度$C$在$(\infty, \alpha_m)$区间内变化。虽然从理论上讲所有的相速度都对某一距离上的响应有所贡献，但根据走时曲线，很明显在有意义的时间－距离范围内相速度可被限制。如图9-2-1所示。

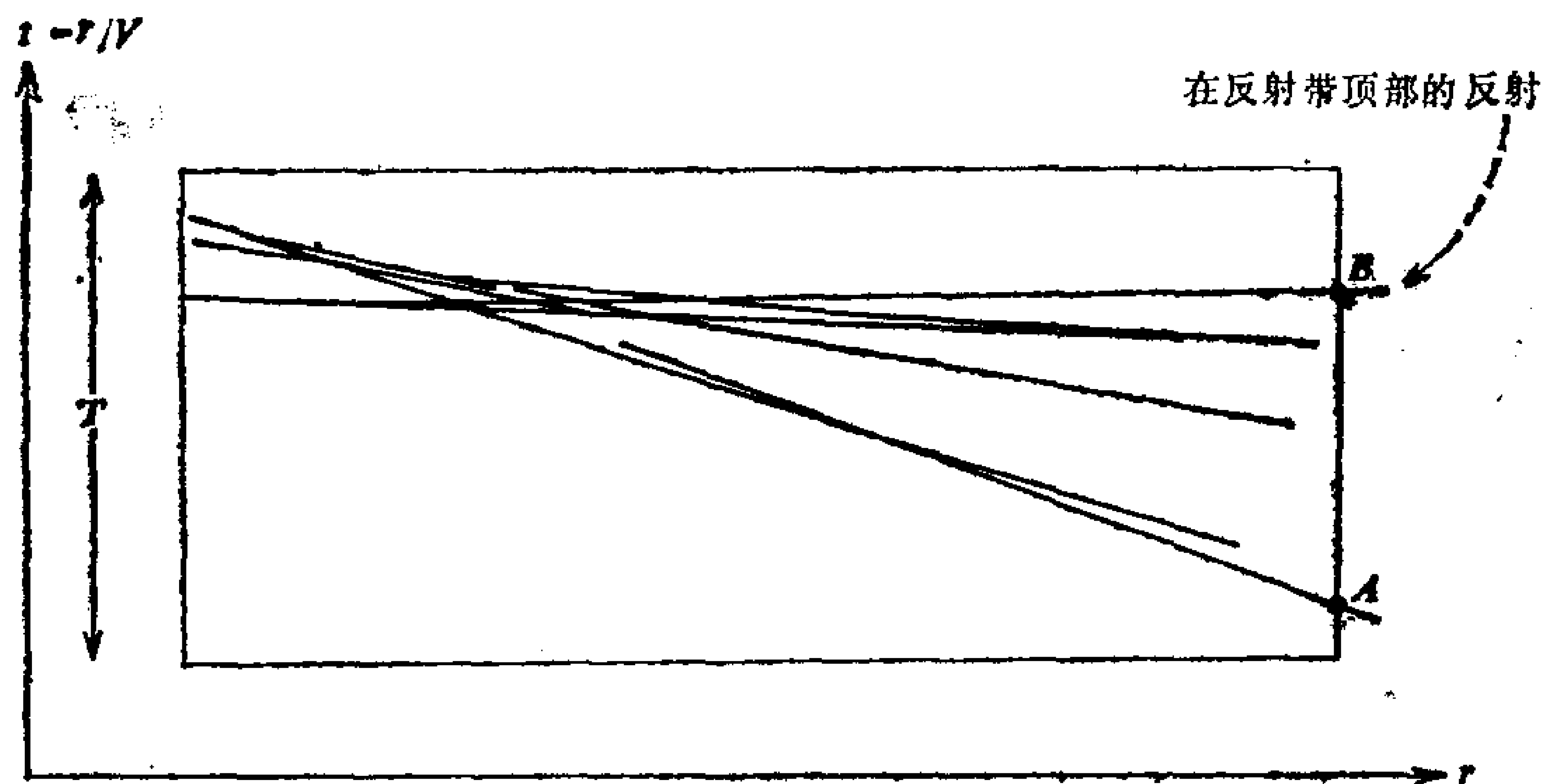



图9-2-1 任意模型的折合走时图

当射线在反射带顶部接近垂直入射时相速度$c_1$最大，当接近于水平入射时相速度$c_2$最小。几乎等于反射带上部覆盖层中的$p$波速度$\alpha_m$。，相应的入射角是

$$\gamma_{1,2} = \sin^{-1}\frac{\alpha_m}{c_{1,2}}. \tag{9-2•6}$$

最小相速度 $c_2$ 的选择要大于覆盖层中的最大 $P$ 波速度。当在大的接收距范围研究反射带顶部的超临界反射时，$c_2$ 要取得尽可能接近覆盖层中的最大$P$ 波速度，一般可取覆盖层中的最大 $P$ 波速度加0.02公里/秒。

**5. 角度窗**

与相速度窗（$c_1$，$c_2$）相应的是角度窗（$\gamma_1$，$\gamma_2$）。在数字积分中，角度增量 $\Delta\gamma$ 要尽可能选得小些，否则会在理论地震图上出现明显的干扰，降低信噪比，甚至会掩盖真正的反射波信号，如图 9-2-2 所示。

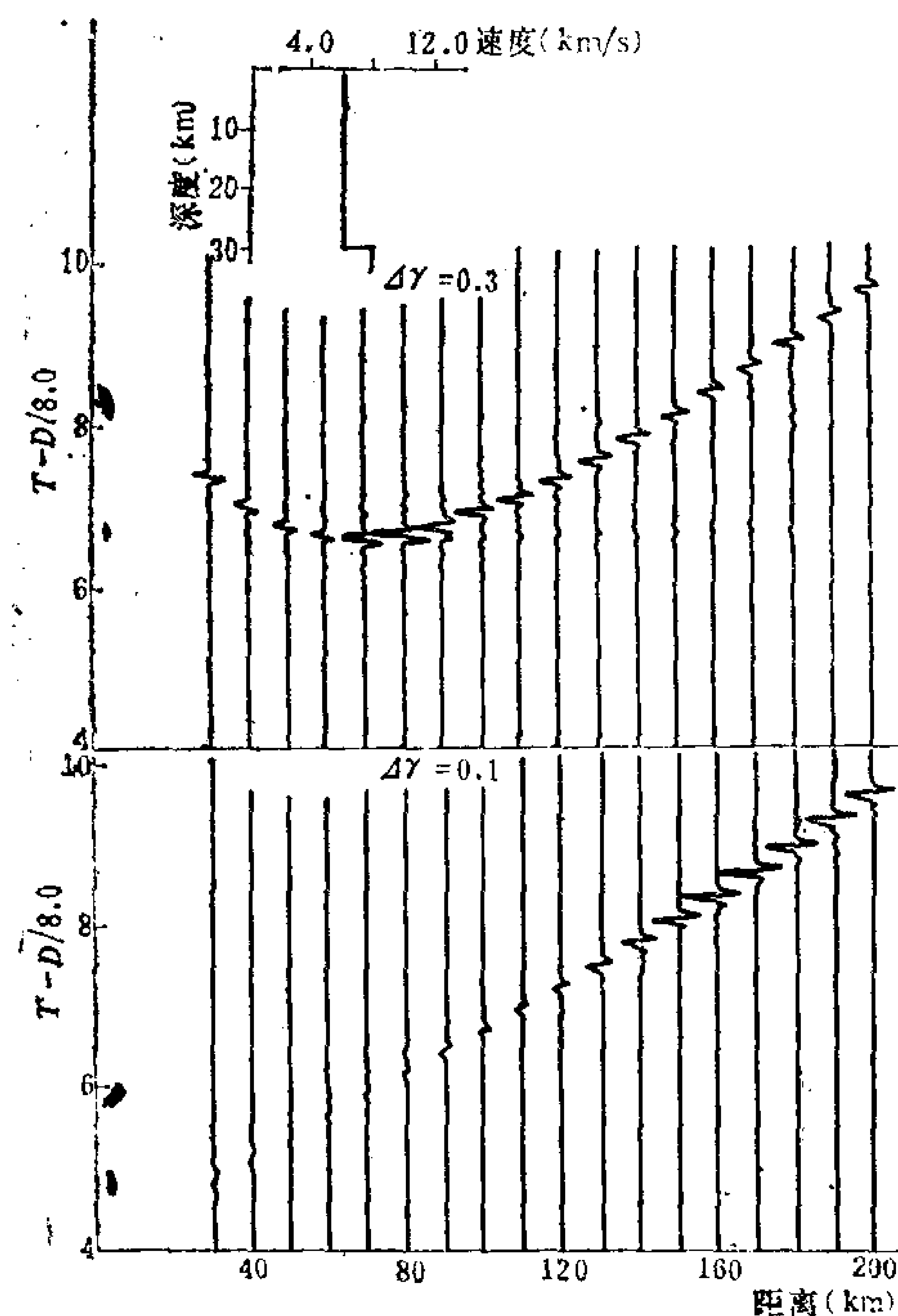



图9-2-2 角度增量的影响

当给定某一频率时，$\Delta\gamma$ 决定了计算反射系数时所需要的角度样点数。因此 $\Delta\gamma$ 取得小，将意味着计算时间的增长。$\Delta\gamma$ 的使用范围一般可在0.05°—0.5°。

对于频率窗和角度窗，我们可以把它们绘制在同一张图上，称为频率-角度窗，如图 9-2-3 所示。

**6. B 类控制方法**

覆盖层和反射带层是连在一起的，反射带的第一个反射界面上部的那一层属于覆盖层。我们可以通过序号的改变使得覆盖层下部与反射层相接触的一层或几层属于反射带，也同样可使反射

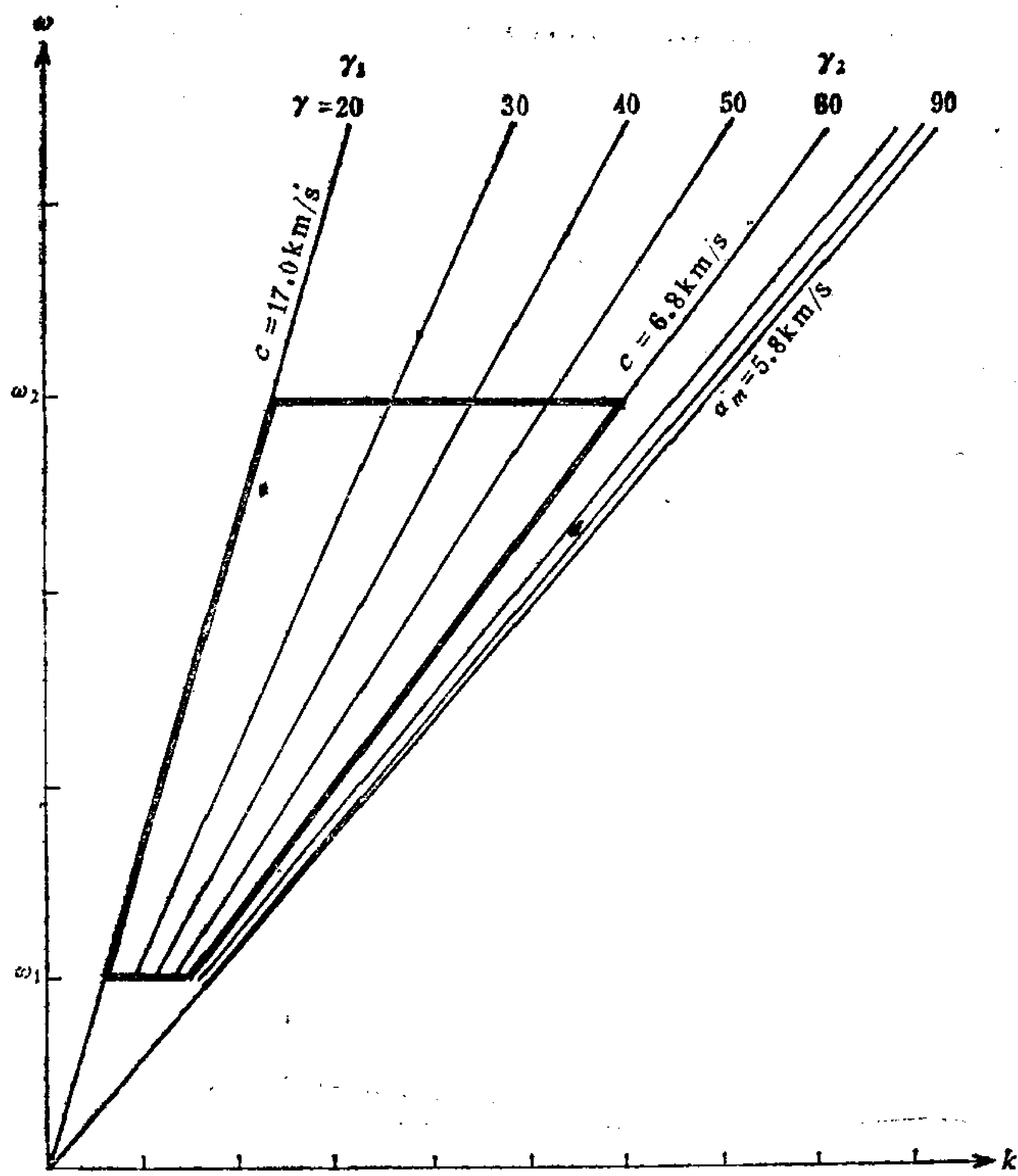



图9-2-3 频率-角度窗

带上部的一层或几层属于覆盖层。在给定的模型中，如果我们只想研究反射带的第一个界面的性质，那么对这个界面下部的各界面可以不予考虑。如果要研究的是下面某个(或几个)界面，可以把这个界面以上的层全部看成覆盖层。这样在计算中，由于求反射系数的Haskell矩阵相乘次数减少，故可缩短计算时间。另一方面由于界面单一，使得记录响应持续时间减少，因此也可缩短计算时间。

当介质层与层之间不是速度差异明显的界面，而是一个过渡

层时，我们可以用一些速度差异较小，厚度很薄的均匀层来代替。当地球曲率不可忽略时，我们还要进行地球改平校正，把球面改成平面。

## （三）震源函数

对于震源函数我们选用爆炸点源，其表达式如下：

$$F(t)=\begin{cases}\frac{1}{\delta}[1-\cos\delta t+\frac{1}{m^2}(\cos m\delta t-1)], & 0\leqslant t\leqslant T;\\ 0, & t<0\text{或}t>T.\end{cases}$$

（9-2•8）

在上式中，$T$ 为震源的持续时间，

$$m=(N+2)/N,$$

$$\delta=N\pi/T,$$

$$N=1,\ 2,\ 3\cdots.$$

其中 $N$ 表示震源函数的极点个数。(9-2.8)式的一阶导数是

$$F'(t)=\begin{cases}\sin\delta t-\frac{1}{m}\sin m\delta t, & 0\leqslant t\leqslant T;\\ 0, & T<0\text{ 或 }t>T.\end{cases}$$

（9-2•9）

$F(t)$ 和 $F'(t)$ 的图形如图9-2-4所示。

## （四）对Bessel函数的处理

对于大的幅角我们使用 Bessel 函数的渐近式：

$$J_0(kr)=\sqrt{\frac{2}{\pi kr}}\cos(kr-\frac{\pi}{4})$$

$$=\frac{1}{\sqrt{2\pi kr}}\left\{\exp[j(kr-\frac{\pi}{4})]+\exp[-j(kr-\frac{\pi}{4})]\right\}$$

（9-2•10）

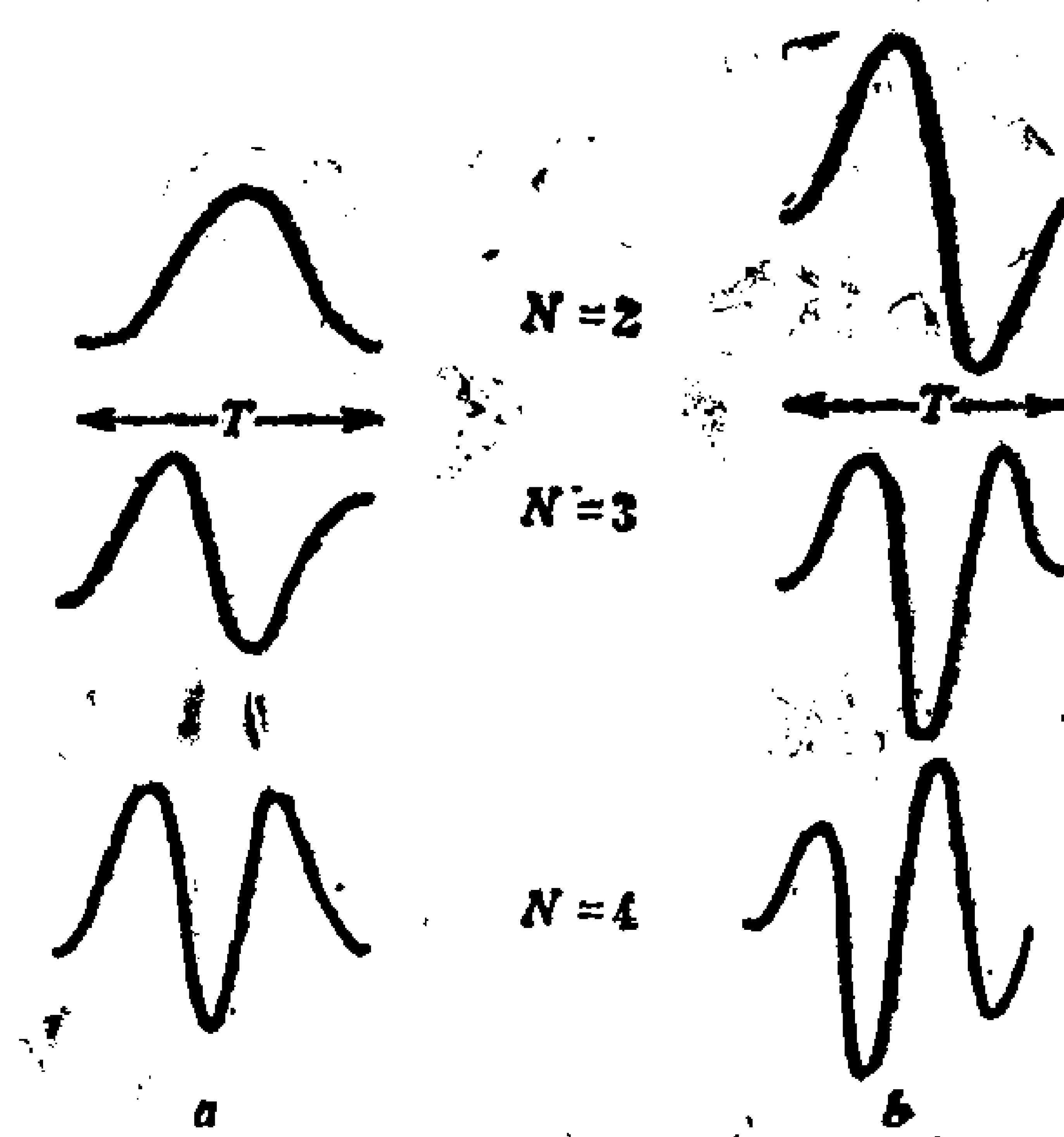



图9-2-4 震源函数和它的导数曲线

$$J_1(kr)=\sqrt{\frac{2}{\pi kr}}\cos(kr-\frac{\pi}{4})$$

$$=\frac{-j}{\sqrt{2\pi kr}}\left\{\exp[j(kr-\frac{\pi}{4})]-\exp[-j(kr-\frac{\pi}{4})]\right\}. \tag{9-2·11}$$

在上面两式中，

$$k=k_{\alpha m}\cdot\sin\gamma,$$

$$k_{\alpha m}=\omega/\alpha_m.$$

$\gamma$ 为射线在反射带顶部的入射角，$\alpha_m$ 为反射带顶部覆盖层中的 $P$ 波速度。

(9-2.10)和(9-2.11)式中的第二项相当于在 $r$ 正方向传播的波(即离源波)，而第一项相当于在 $r$ 负方向传播的波(即向源波)。在远距离范围内只有外行波有物理意义，而向着震源传播的波可以忽略。这里需要注意的是 $kr$ 不可取得过小，否则 Bessel 函数的渐近式失去意义。实践证明当 $kr>15$ 时，(9-

2.10)和(9-2.11)式所表示的渐近式是有效的。

### (五) 积 分 方 法

数字积分用 Simpson 法 或者用梯形法去做。梯形法产生的干扰比 Simpson 法小，这一点已为一些研究者所证明（Bakun，1970）。

## §9-3 用广义射线理论计算理论地震图的一般原理

前面我们介绍了计算理论地震图的反射率法，本节将介绍计算理论地震图的另一种方法——广义射线理论法。这种理论与反射率法在计算方法上有着本质的差别，但在理论上却是一致的。在这一节我们将介绍扩展后 的 Cagniad（Cagniad，L.，1939）方法。这种方法使用由许多均匀水平界面构成的介质模型。在计算方法上，它使用复射线参数平面，并在其上得到 Cagniad 积分路径。对于速度随深度变化的球对称介质，已经证明（Andrianova 等，1967）*SH* 波能从平面层状介质直接得到球型介质的精确解。而对于 *P* 波和 *SV* 波获得这样的精确解似乎是不可能的（Chapman，1973），因而只能使用地球改平变换把球层改变成平面层，把随深度变化的梯度层变成叠加在一起的均匀薄层。这种问题对于反射率法同样存在。

假定介质模型和射线路径如图9-3-1所示。其中：$\alpha_i$，$\beta_i$，$\rho_i$，$d_i$ 分别表示第 $i$ 层的纵波速度，横波速度，介质密度和层的厚度。我们还假定位于 $r=0$，$z=0$ 处的点源为单位阶梯函数，其表达式是：

$$\varphi_1(r,\ z,\ t)=H(t-R/\alpha_1)/R \qquad (9\text{-}3\cdot1)$$

其中 $H$ 为 Heaviside 单位函数，当 $\tau\leqslant0$ 时，$H(\tau)=0$，当 $\tau>0$，

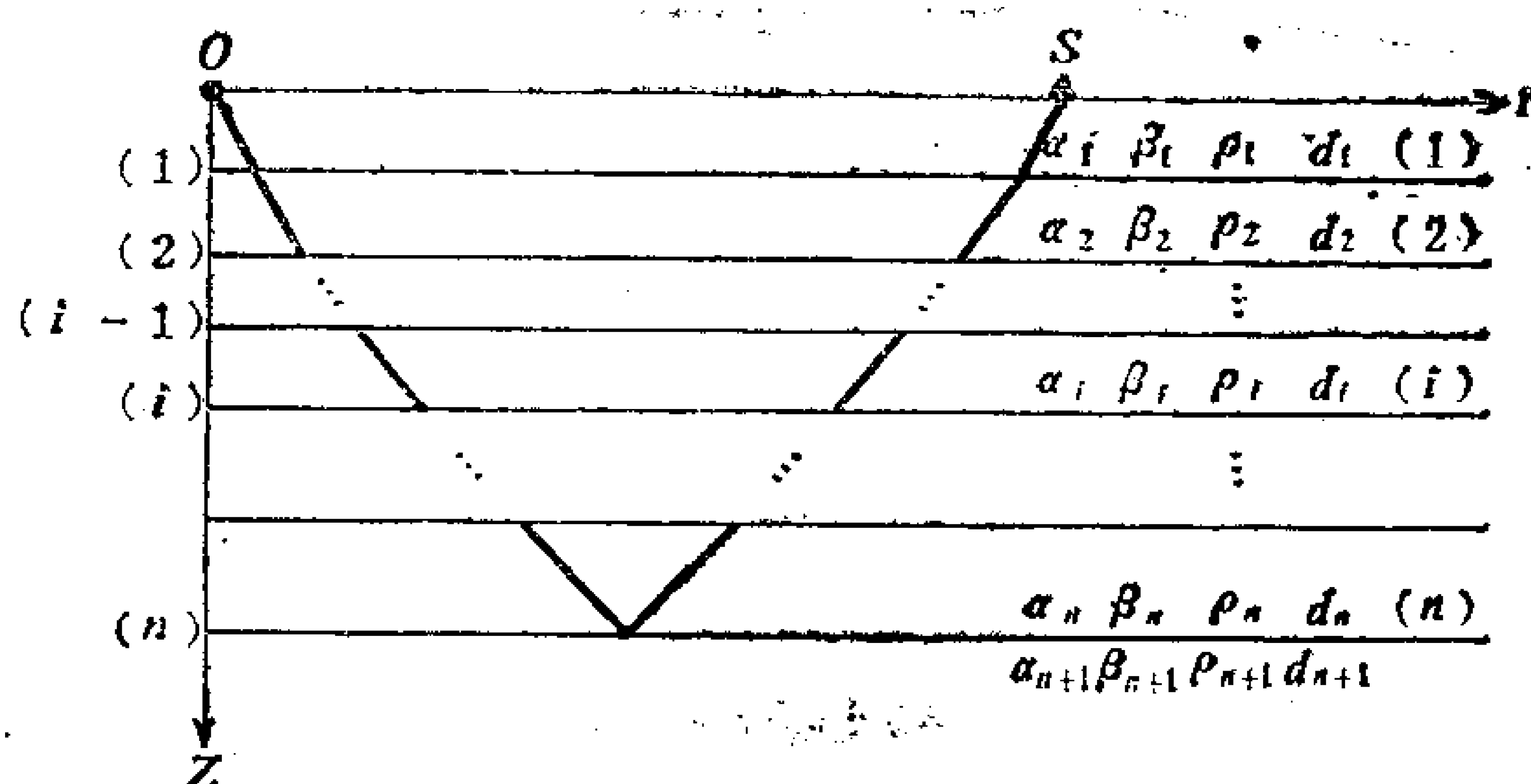



图 9-3-1

当$\tau>0$时，$H(\tau)=1$，$R=\sqrt{r^2+z^2}$，表示离开震源的距离，对(9-3.1)进行Laplace 变换得到

$$\overline{\varphi}_1(r,z,s)=\int_0 J_0(kr)e^{-(k^2+\frac{s^2}{\alpha^2})^{1/2}z}\frac{k}{\sqrt{k^2+s^2/\alpha^2}}dk$$

(9-3·2)

式中取 $k=-jsp$。于是有：

$$\overline{\varphi}_1(r,\ z,\ s)=\frac{2s}{\pi}\mathrm{Im}\int_0^{j}\mathrm{K}_0(spr)e^{-s\eta_1 z}\frac{p}{\eta_1}dp,\qquad (9\text{-}3\cdot 3)$$

在这里 $j$ 表示虚数单位，$\mathrm{K}_0$ 为修正 Bessel 函数 $\eta_1=(\alpha_1^{-2}-p^2)^{1/2}$。

由震源发出的射线通过图9-3-1所示的介质模型并在第 $n$ 个界面上发生反射，我们在地表距离震源$r$处所接收到的$p$波位函数的Laplace 变换是：

$$\overline{\varphi}(r,\ z,\ s)=\frac{2s}{\pi}\mathrm{Im}\int_0^{j}\mathrm{K}_0(spr)T_d\cdot R_{PP}\cdot T_u\exp\left[zs\sum_{i=1}^{n}d_i\eta_i\right]$$

$$\times\frac{p}{\eta_1}dp.\qquad (9\text{-}3\cdot 4)$$

式中

$$T_d = T_{d1} \cdot T_{d2} \cdot \cdots \cdot T_{di} \cdot \cdots \cdot T_{dn-2} \cdot T_{dn-1}, \qquad (9\text{-}3\cdot 5)$$

其中 $T_{di}$ 表示射线向下传播，穿过第 $i$ 个界面的透射系数。

$$T_u = T_{n-1} T_{u-2} \cdots T_{ui} \cdots T_{u2} T_{u1}, \qquad (9\text{-}3\cdot 6)$$

$T_{ui}$ 表示射线向上传播穿过第 $i$ 个界面的透射系数，$R_{PP}$ 表示第 $n$ 个界面的反射系数。$\eta_i = (a_i^{-2} - p^2)^{1/2}$，积分时以 $\mathrm{Re}\,\eta_i \geqslant 0$ 选取割线。在(9-3.4)中被积函数包括了平面波的反射系数和透射系数。这些系数在通常情况下很容易写出来。

为了简化计算，我们把 $k_0$ 展开成渐近式，

$$K_0(spr) = \left(\frac{\pi}{2spr}\right)^{1/2} e^{-spr}\left\{1 - \frac{1}{8spr} + 0[(spr)^{-2}]\right\}.$$

取其中第一项则有：

$$K_0(spr) = \left(\frac{\pi}{2spr}\right)^{1/2} e^{-spr}.$$

把上式代入(9-3.4)，于是有：

$$\varphi(r, z, s) = \left(\frac{2s}{\pi r}\right)^{1/2} \mathrm{Im}\int_0^{j\infty} \mathscr{B}(Sp)\exp\left\{-s\left(pr + \sum_{i=1}^{n} d_i \eta_i\right)\right\} \cdot \frac{\sqrt{p}}{\eta_1} dp, \qquad (9\text{-}3\cdot 7)$$

式中 $\mathscr{B}(s,p) = T_d R_{PP} T_u$。在(9-3.7)中我们取

$$\tau = pr + \sum_{i=1}^{n} d_i \eta_i, \qquad (9\text{-}3\cdot 8)$$

反演(9-3.7)的Cagniad路径是(9-3.8)的解，$p = p(\tau)$，这里 $\tau$ 为

正实数。(9-3.7)的反演表达式是

$$\varphi(r,\ z,\ t)=\frac{2}{\pi}\mathrm{Im}\int_0^t\frac{p}{\eta_1}\left(\frac{dp}{d\tau}\right)\frac{\mathscr{B}(p)H(t-\tau)d\tau}{(t-\tau)^{1/2}\ (t-\tau+2pr)^{1/2}}.$$

(9-3·9)

利用单位阶梯函数的性质，即参量必须为正实数，(9-3.9) 又可写成

$$\varphi(r,\ z,\ t)=\frac{2}{\pi}\mathrm{Im}\int_0^t\frac{p}{\eta_1}\left(\frac{dp}{d\tau}\right)\frac{\mathscr{B}(p)d\tau}{(t-\tau)^{1/2}\ (t-\tau+2pr)^{1/2}}.$$

(9-3·10)

当震源函数频率较高，持续时间较短时，$t-\tau+2pr\approx 2pr$。于是(9-3.10)可写成褶积形式

$$\varphi(r,\ z,\ t)=\frac{2}{\pi}\frac{1}{\sqrt{t}}*\Psi(t)$$

其中 $$\Psi(t)=\mathrm{Im}\left[\frac{\sqrt{p}}{\eta_1}\ \frac{dp}{d\tau}\cdot\frac{\mathscr{B}(p)}{\sqrt{2r}}\right] \qquad (9\text{-}3\cdot 11)$$

在实际的资料处理过程中，我们往往使用更一般的震源函数 $F(t)$ 而不是单位阶梯函数 $H(t)$，同时还要考虑接收仪器的频率响应 $s(t)$。这样，我们还需要对 $F(t)$，$s(t)$ 等进行褶积运算。至于地面的反射影响，可以按照在反射率法中叙述的方法处理。用广义射线理论法计算的理论地震图如图9-3-2所示。

还有一种常用的计算理论地震图的方法，人们称其为全波理论。这种方法可以直接用于球对称介质模型。该理论假定在任意厚度的非均匀层中P 波和SU波可以分离，从而使用近似，并且使用一致渐近序列的第一项表示辐射本征函数。关于这种方法的基本原理，本章不再叙述，读者可参看由 Aki 和 Richards(1980) 合著的定量地震学的第九章。

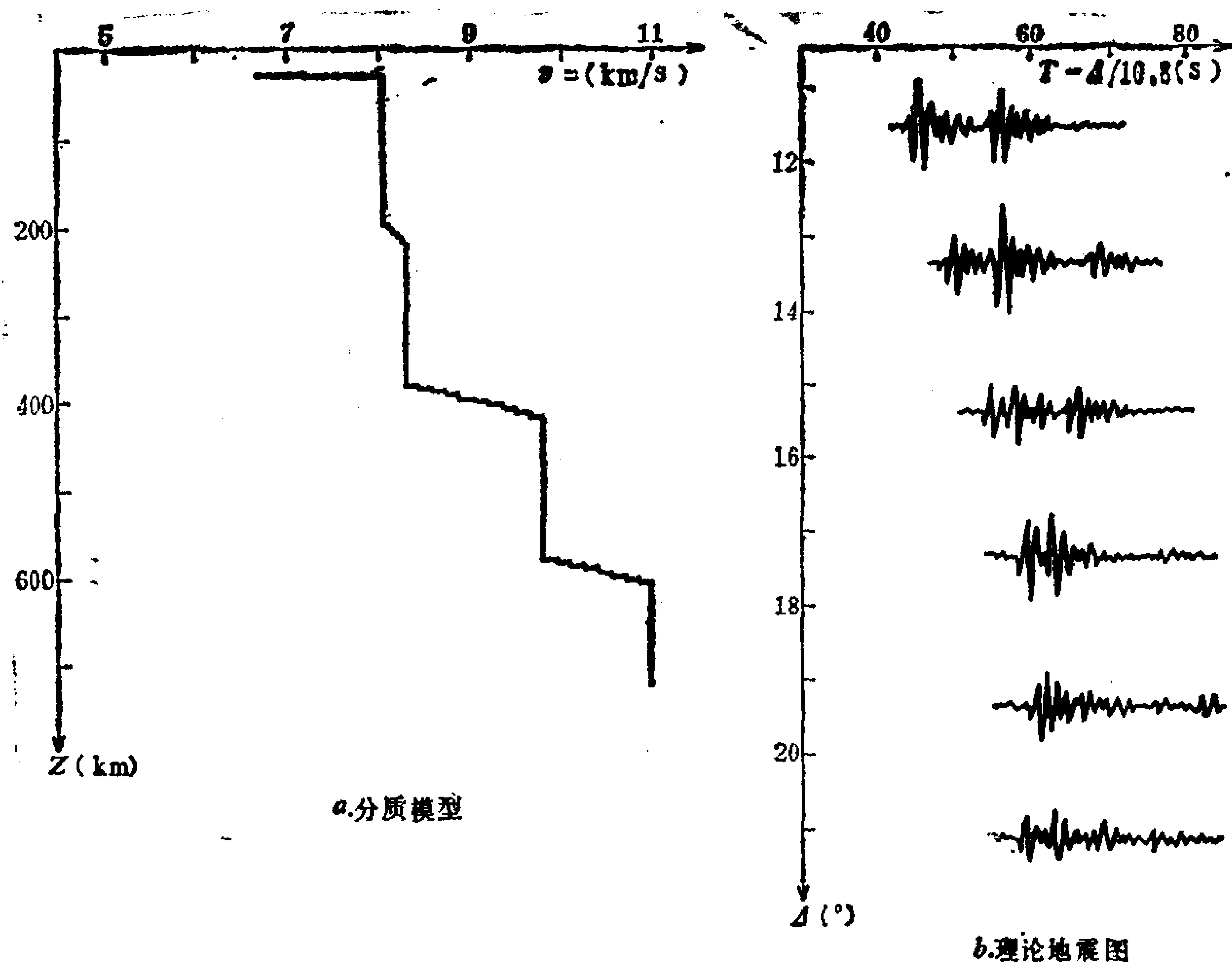



图 9-3-2

## §9-4 理论地震图与观测记录的比较

在这一节我们把用反射率法计算的理论地震图与实际的观测记录相比较。图 9-4-1 是华北某剖面的部分观测记录。虽然我们不知道这些记录的放大倍数，仪器特性及震源辐射的脉冲形状，但我们可以通过模拟记录图的某些最突出的振幅特性，获得对该剖面的走时曲线的解释，并由此对地下介质分布进行探讨。当然我们在这里给出的图形比较仅仅作为一个例子，证明这种研究的可能性。

图9-4-2是用反射率法计算的理论地震图。图中的计算模型由

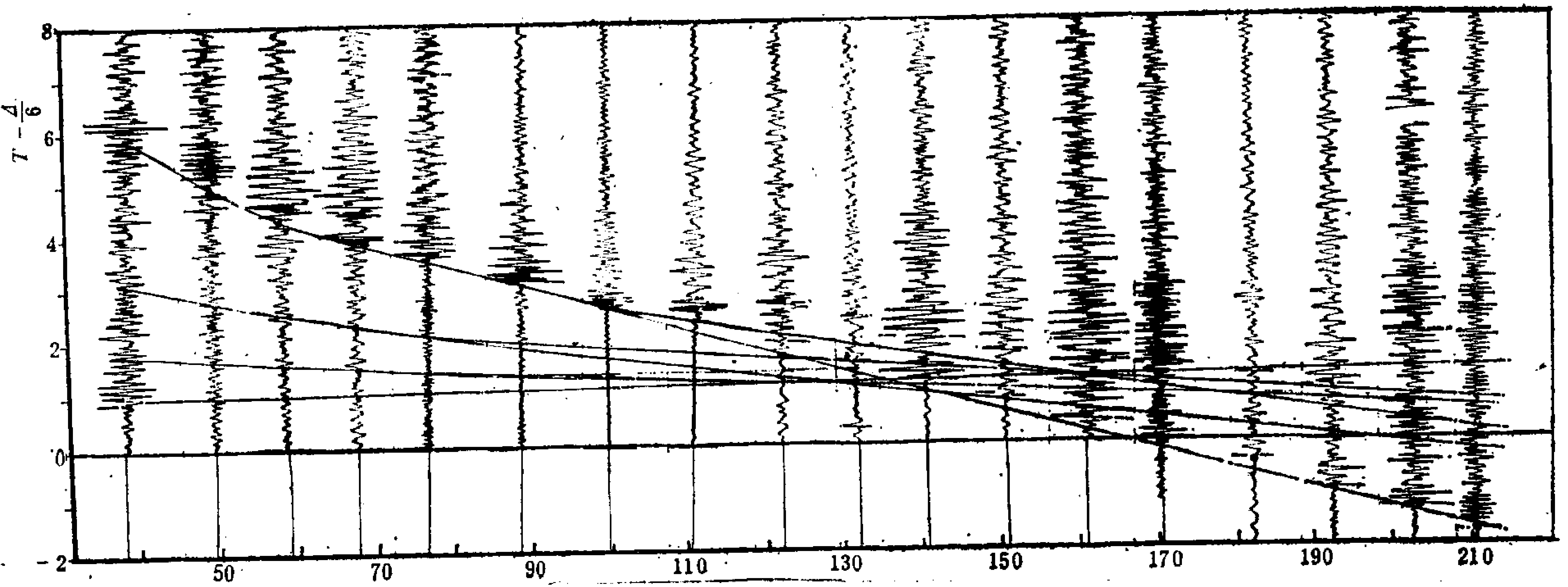
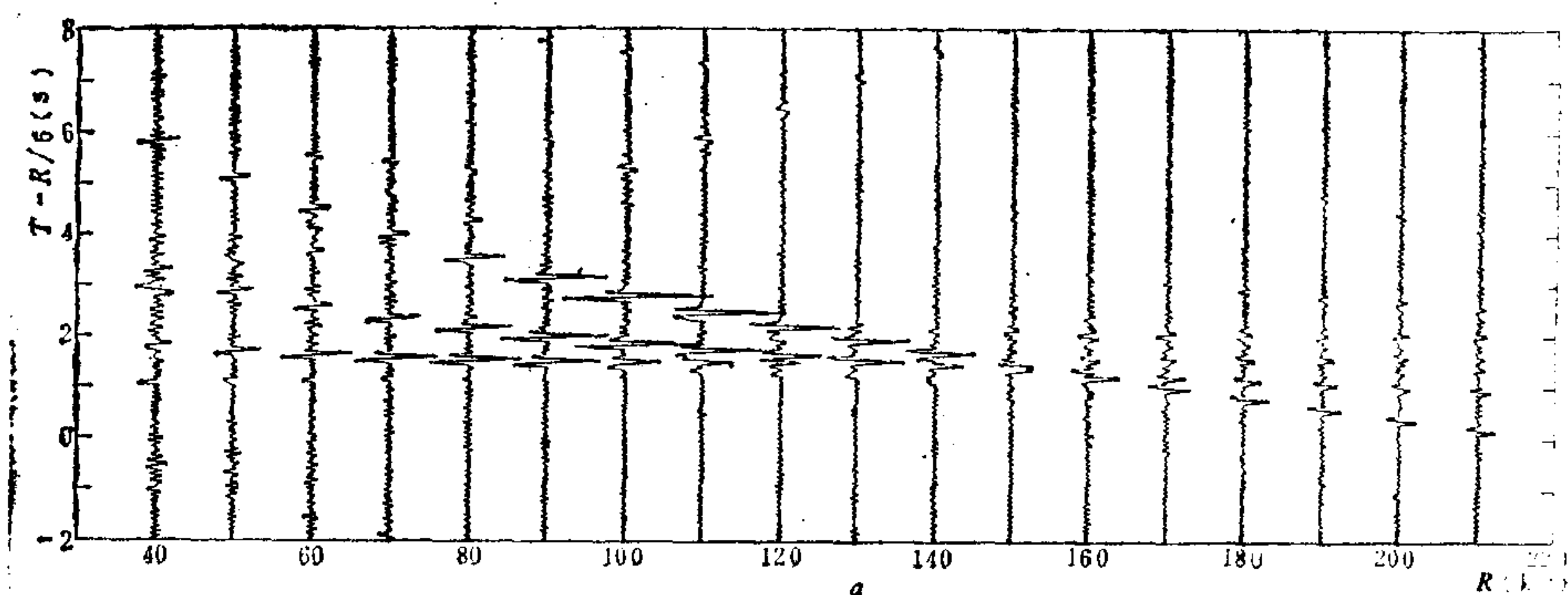



图9-4-1 观测记录



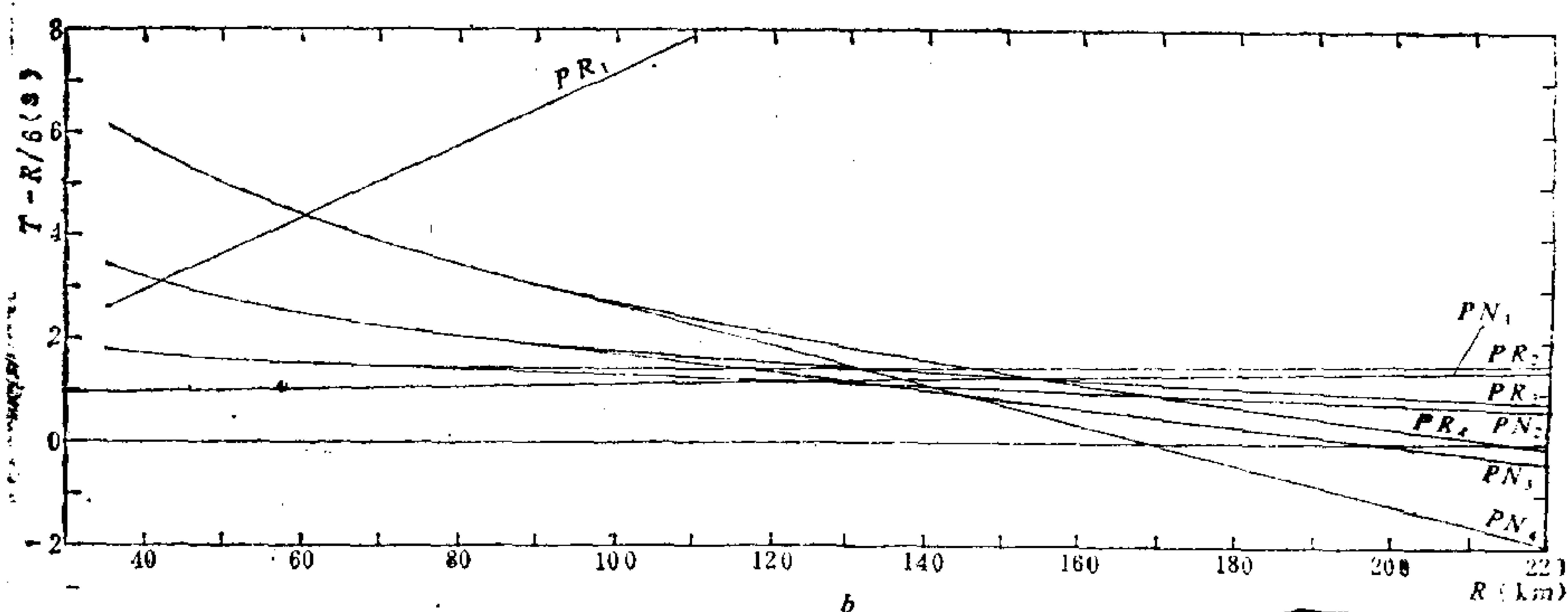




图9-4-2 理论模型及合成记录

a.理论地震图； b.时间-距离曲线

分层均匀各向同性完全弹性的层状介质组成。在计算理论地震图时，除图中标明的参数外,其它参数的 选 择是：$T$ 取10.24s，$\Delta t$ 取0.02s，$\Delta \gamma$取0.14°，频带宽为15Hz，相速度窗为(4.22，30)，角度窗为(8.19°,77.35°)。图中的干扰主要是由于$\Delta \gamma$选择得偏大引起，如果把$\Delta \gamma$取成0.1度,则可进一步压制干扰，提高信噪比。

我们把图9-4-2与图9-4-1进行比较，发现相应震相所构成的走时曲线有着很好的一致性，但在动力学特征上却有着明显的

差异。所谓动力学特征，我们指的是同道记录的各震相的振幅比。（因为理论地震图与观测记录的震相振幅之间不能直接进行比较）。通过比较我们发现相应于莫霍界面的理论首波振幅偏小，而相应于康腊界面及玄武岩层内界面的理论反射波振幅却偏大。这些都可能是由于我们使用了带有明显速度差异的界面所引起的。

从图 9-4-2，我们还可以看到这样一种现象，反射波的最大振幅出现在临界反射距离之后的某一距离上，这一点已在理论上有所证明（červeny，1961）。

## §9-5 理论地震图计算程序

下面我们给出反射率法理论地震图的计算程序。首先对程序中主要标识符及程序结构说明如下。

### （一）主要标识符ISS（P）——控制变量数组

M——覆盖层的层数；
Z(I)——界面埋藏深度；
D(I)——层的厚度；
A(I)——P 波速度；
B(I)——S 波速度；
RHO(I)——介质密度；
X1——初始距离；
X2——终点距离；
DX——距离增量；
VRED——折合速度；
TMIN——最小折合时间；
DGAM——角度增量
IDFEN——用于角度窗的整型量，IDFEN*DGAM 表示角

度左边窗的长度，ISS(24) * IDFEN * DGAM 表示角度右边窗的长度；

C1——最大视速度；

C2——最小视速度；

FU——频率下限；

FWIL——从 FU 到 FWIL 为左频率窗；

FO——频率上限；

FWIR——从 FWIR 到 FO 为右频率窗；

DT——时间增量；

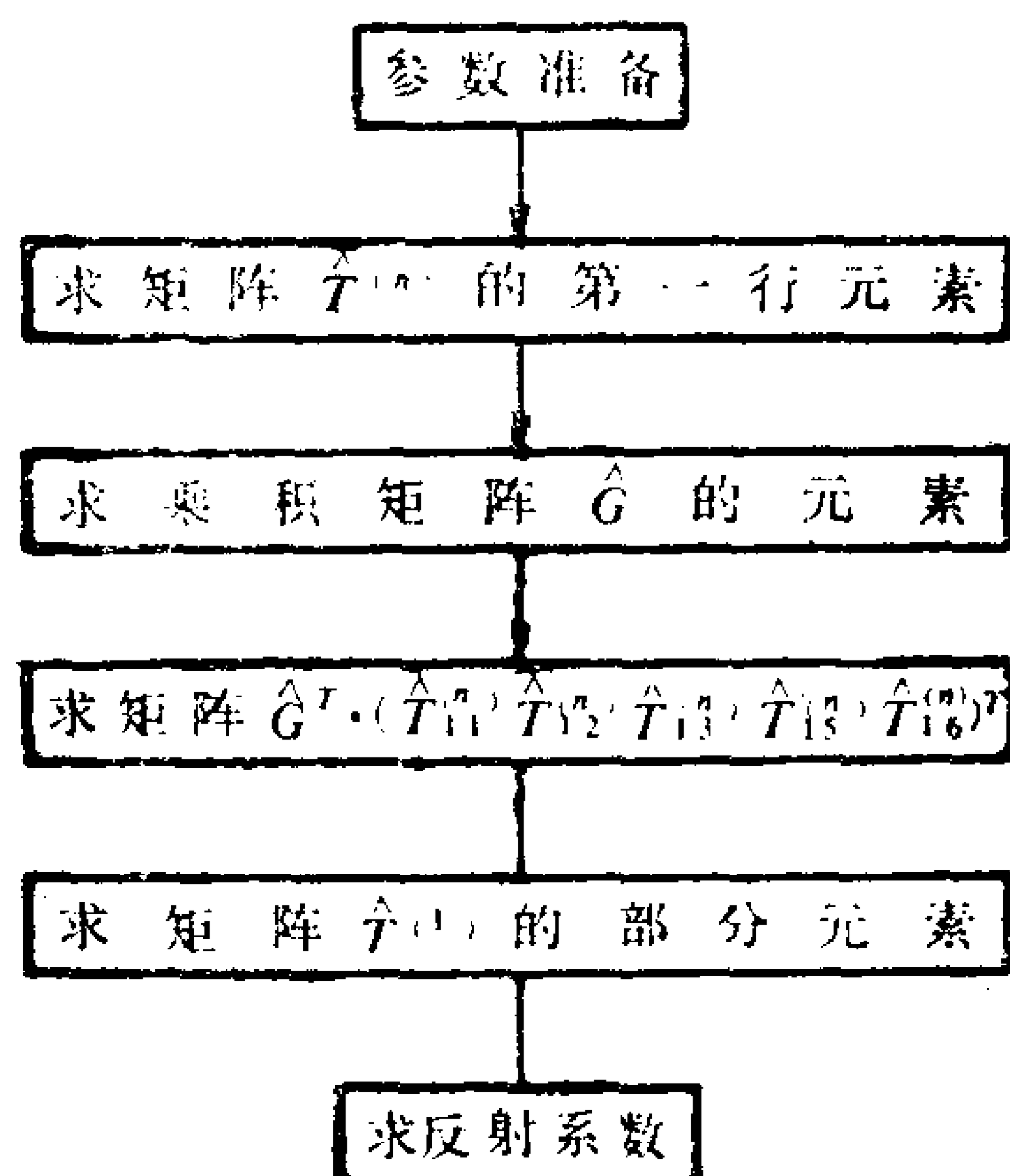
NPTS——时间样点数；

NA——信号源前面的零点个数；

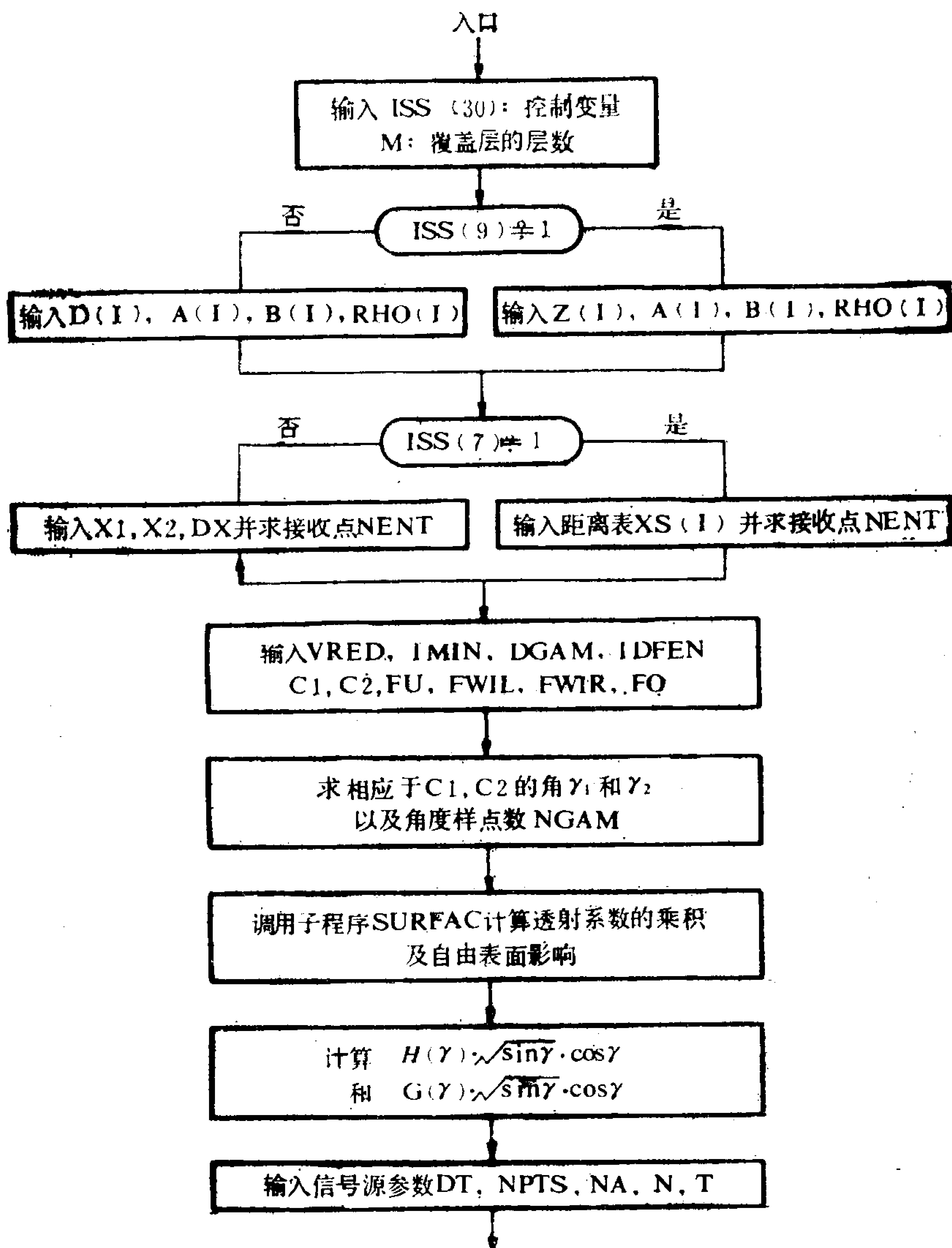
N——信号源点的极点个数。

## （二）程 序 框 图

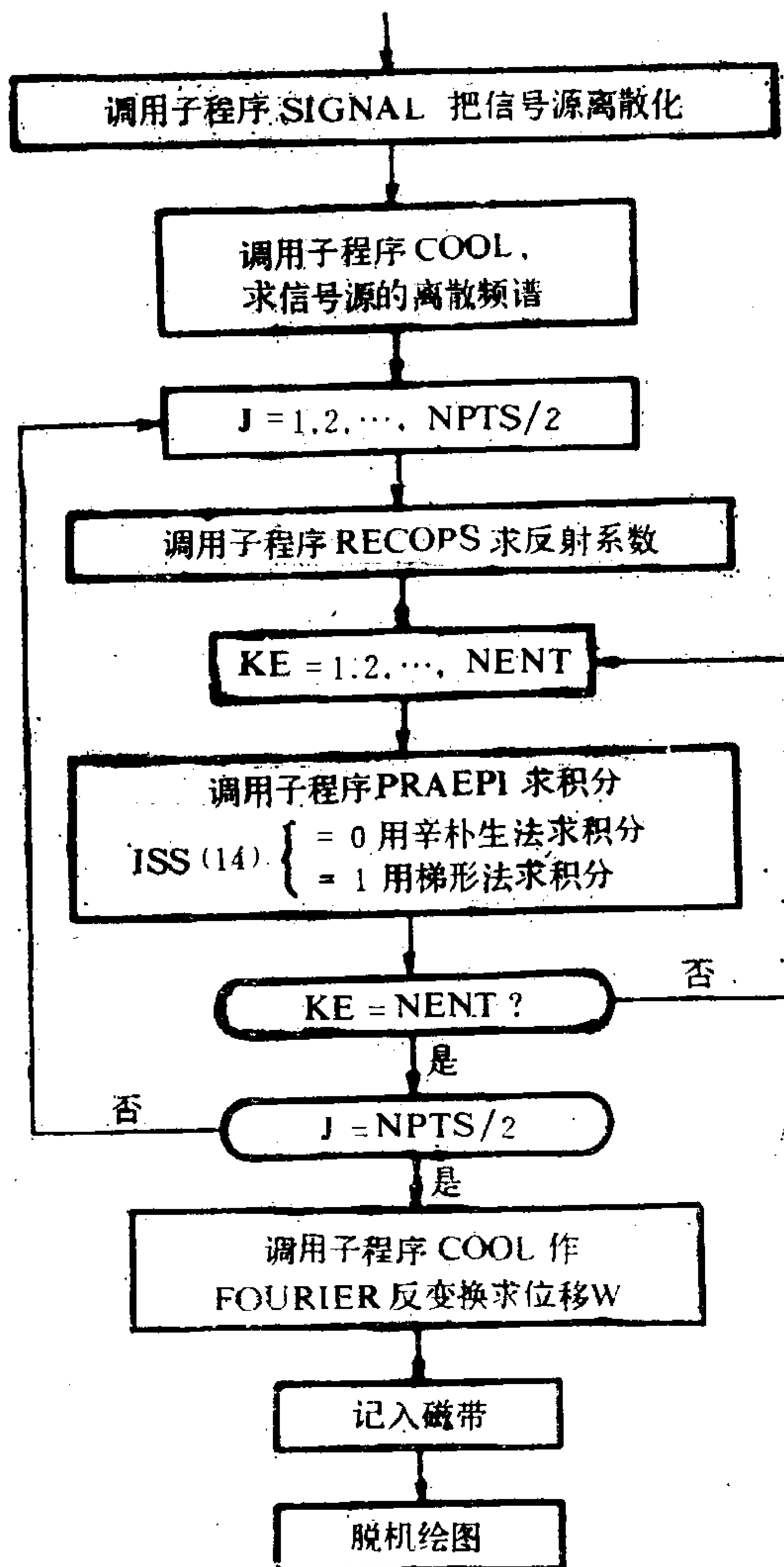
### 1. 子程序 RECOPS 框图



## 2. 主程序框图



入口
输入 ISS（30）：控制变量
M：覆盖层的层数
否
ISS（9）≠1
是
输入D（I），A（I），B（I），RHO（I）
输入Z（I），A（I），B（I），RHO（I）
否
ISS（7）≠1
是
输入X1，X2，DX并求接收点NENT
输入距离表XS（I）并求接收点NENT
输入VRED，IMIN，DGAM，IDFEN
C1，C2，FU，FWIL，FWIR，FO
求相应于C1，C2的角γ1和γ2
以及角度样点数NGAM
调用子程序SURFAC计算透射系数的乘积
及自由表面影响
计算 H（γ）·√sinγ·cosγ
和 G（γ）·√sinγ·cosγ
输入信号源参数DT，NPTS，NA，N，T



调用子程序 SIGNAL 把信号源离散化
调用子程序 COOL，
求信号源的离散频谱
J =1,2,…, NPTS/2
调用子程序 RECOPS 求反射系数
KE =1,2,…, NENT
调用子程序PRAEPI 求积分
ISS(14) = 0 用辛朴生法求积分
= 1 用梯形法求积分
KE =NENT ?
否
是
J =NPTS/2
否
是
调用子程序 COOL 作
FOURIER 反变换求位移W
记入磁带
脱机绘图

## （三）程序结构

程序分主程序和子程序两部分。其中子程序部分包括十四个子程序和两个函数段。下面给出这些程序间的调用关系及各程序的作用。

1．子程序的作用

（1）子程序 PRAEPI 和子程序 INTGRA的总体作用是做数字积分。

（2）子程序 PHASE 计算 Bessel 函数。在程序中分两种情况，当ISS(1)＝1时，用三角函数计算，否则用指数计算。

（3）子程序 SURFAC 计算透射系数和自由表面影响函数的乘积。

（4）子程序 RCR 求首波临界距离。

（5）子程序 RECOPS 求反射系数 $R_{PP}$，同时还可求出反射系数 $R_{PS}$，$R_{SS}$ 和 $R_{SP}$。

（6）子程序 SIGNAL 把信号源离散化。

（7）子程序 COOL 可做正反 Fourier 变换。

（8）子程序 ZLOG 求指数。

（9）子程序 MAXSN 求最大 $P$ 波速度值。

（10）FUNCTION AMAX 和 FUNCTION AMIN 分别求最大值和最小值。

2．程序调用关系

主程序
- ①CALL RCR→CALL MAXSN
- ②CALL SURFAC
- ③CALL SIGNAL
- ④CALL ZLOG
- ⑤CALL COOL
- ⑥CALL RECOPS
- ⑦CALL PRAFPI→CALL INTGRA→CALL PHASE
- ⑧引用 FUNCTION AMAX
- ⑨引用 FUNCTION AMIN

（四）控制变量的意义（只说明程序中现用的变量）

| I | ISS(I) | |
|---|---|---|
| 1 | = 0 | 用渐近式 $\exp\{-j(x-\pi/4)\}/\sqrt{2\pi kr}$ 代替 $J^0(kr)$；$J_1(kr)$ 类似。 |
| | = 1 | 用 $\cos(kr-\pi/4)*\sqrt{2/\pi kr}$ 代替 $J_0(kr)$，$J_1(kr)$ 类似。 |
| 4 | = 0 | 计算垂直位移和水平位移。 |
| | = 1 | 只计算垂直位移。 |
| | = 2 | 只计算水平位移。 |
| 5 | = 0 | 用解析表达式给出信号源。 |
| 7 | = 0 | 输入最小距离，最大距离和距离增量。 |
| | = 1 | 用读卡机读入距离表。 |
| 9 | = 0 | 读入层参数：层的厚度，$P$ 波和 $S$ 波速度，介质密度。 |
| | = 1 | 读入层参数：界面深度，$P$ 波和 $S$ 波速度，介质密度。 |
| 12 | = 0 | 数据记带准备绘图。 |
| 13 | = 1 | 当 $S$ 波速度等于零时，用 $V_S=V_P/\sqrt{3}$ 计算 $S$ 波速度。 |
| 14 | = 0 | 用 Simpson 法积分。 |
| | = 1 | 用梯形法积分。 |
| 18 | = 0 | 无角度窗。 |
| | = 1，2，… | 角度两边窗。 |
| 19 | = 0 | 无频率窗。 |
| | = 1，2… | 频率两窗边。 |

## （五）程序使用

程序使用分计算和绘图两部分。在计算部分中输入的参数

有：

（1）控制变量 ISS(I)，覆盖层数 $M$。

（2）模型参数 A(I),B(I),RHO(I) 以及 D(I)(或 Z(I))。

（3）输入接收距 X。对接收距的输入可分两种情况，一是等间距输入，即输入初始距离 X1，终点距离 X2，步长DX；二是输入距离表，这可用于非等间距。具体采用哪种输入方式，在程序中用 ISS(7) 控制。

（4）输入折合速度 $V_{RED}$ 和最小折合时间 TMIN。

（5）输入求 $\gamma$ 角及角度样点数的有关参数：DGAM，IDFEN，C1，C2，FU，FWIL，FWIR，FO。

（6）输入信号源参数 DT，NPTS，NA，N，T。绘图程序（略）。

## （六）计 算 程 序

本程序是用标准 FORTRAN-Ⅳ 语言编写，对于使用扩充 FORTRAN 语言的计算机也可应用。源程序内容如下：

```
C          THE MAIN PROGRAM
           COMMON S(512), CPP(800), CPS(800), Z(100),
              D(100), A(100),
              B(100)RHO(100), X(30), SINA(800),COSA(800),
              PI2, DT, DAUER,
              FREQ, DGAM, C1, C2, PPR, PPI, TI, TO,
              ISS(30),INPT, NG1, NG2, NOL,
              NPTS,NENT,NGAM, DADA(800), DIDI(800), M
          COMPLEX S, CPP, CPS
          DIMENSION XS(10), BALLA(512)
          DIMENSION WAVEL(512), W(30, 512)
          COMPLEX WAVEL, W
          COMPLEX CU, CV, EINI
          EQUIVALENCE(CPP(1), BALLA(1))
```

```
      OPEN(16, FILE='DIZ', STATUS='UNKNOWN',
         MRECL=2000, BLOCK=2000)
      EINI=(0.,+1.0)
      PI2=2.0*3.1415926
      GABO=PI2/360.
      AMP=0
      WRITE(6, 9000)
9000  FORMAT(1H1, 30X, 'COMPUTATION OF
         THEORETICAL SEISMOGRAM'/46X,
         'BY THE'/38X, 'REFLECTIVITY   METHOD'/)
      READ(5, 9010) (ISS(I), I=1,30), M
9010  FORMAT(30I1, I3)
      WRITE(6,9020) (ISS(I), I=1,30),M
9020  FORMAT(1H0, 34X, 'SWITCH LIST'/10X, 6(5I1,
         3X)/10X, 'NUMBER OF LAYERS
         IN OVERBURDEN', 5X, I3)
      IF(ISS(9).EQ.1) GO TO 1230
      I=0
1205  I=I+1
      READ(5, 9040)D(I), A(I), B(I), RHO(I)
9040  FORMAT(4F10.4)
      IF(ISS(13).EQ.1)  B(I)=A(I)/1.732
      IF(RHO(I).EQ.0.)RHO(I)=0.25+0.3788*A(I)
      IF(D(I)+A(I)+B(I)) 1205, 1210, 1205
1210  NOL=I-1
      Z(1)=D(1)
      N1=NOL-1
      IF(N1.EQ.1) GO TO 1220
      DO 1215 I=2, N1
1215  Z(I)=Z(I-1)+D(I)
1220  GO TO 1250
1230  I=0
```

```
1232  I=I+1
      READ(5, 9040)Z(I), A(I), B(I), RHO(I)
      IF(ISS(13).EQ.1)  B  (I)=A(I)/1.732
      IF(RHO(I).EQ.0.)RHO(I)=0.25+0.3788 * A(I)
      IF(Z(I)+A(I)+B(I)) 1232, 1240, 1232
1240  NOL=I-1
      D(1)=Z(1)
      N1=NOL-1
      IF(N1.EQ.1) GO TO 1248
      DO 1246 I=2, N1
1246  D(I)=Z(I)-Z(I-1)
1248  CONTINUE
C     PARAMETER OUTPUT
C
1250  WRITE(6, 9060)
9060  FORMAT(1H0, 40X, 'PARAMETER LIST'//)
      WRITE(6, 9070)
9070  FORMAT(1H0, 20X, 'DEPTH    THIDKNESS
         P-VELOCITY          S-VELOCITY
         DENSITY'/23X, 2HKM, 12X, 2HKM, 12X,4HKM
         /S, 11X, 4HKM/S, 9X, 7HG/CM
         2* * 3//)
      DO 1260 I=1, N1
      IF(I.GT.M)  GO TO  1255
      WRITE(6, 9080)  Z(I), D(I), A(I), B(I), RHO(I)
9080  FORMAT (11X, 5F15.4)
      GO TO 1260
1255  WRITE(6, 9090) Z(I), D(I), A(I), B(I), RHO(I)
9090  FORMAT (11X, 5F15.4, 5X, 'REFLECTED ZONE')
1260  CONTINUE
      Z(NOL)=9999.999
      D(NOL)=9999.999
```

```
      WRITE(6, 9100) A(NOL), B(NOL), RHO    (NOL)
 9100 FORMAT (41X, 3F15.4, 5X, 'REFLECTED ZONE')
      V1=A(1)
      CALL RCR(D, A, NOL, TI, VMAX)
      IF(ISS(7).EQ.1) GOTO 1800
      READ(5, 9120)  X1, X2, DX
 9120 FORMAT(3F10.4)
      WRITE(6, 9130)  X1, X2, DX
 9130 FORMAT(////, 11X, 'EQUAL INTERVAL DISTA-
         NCE  LEST'//21X, 'X1=', F9.3,
         'KM X2=', F9.3, 'KM DX=', F9.3, 'KM'/)
      NENT=(X2-X1)/DX+1.0
      X(1)=X1
      DO 1710 I=2, NENT
 1710 X(I)=X(I-1)+DX
      GO TO 1900
C      ISS(7)=1 LIST OF NOT EQUAL INTERVAL
C     DISTANCE
C     *
 1800 DO 1810 J=1, 50, 8
      JDC=J+7
      READ(5, 9140) (XS(I), I=1, 8)
 9140 FORMAT(8F10.3)
      DO 1808 I=J, JDC
      IJK=I-J+1
      IF(XS(IJK)) 1806, 1808, 1808
 1806 NENT=I-1
      GO TO 1900
 1808 X(I)=XS(IJK)
 1810 CONTINUE
C     * * * LIST OF DISTANCE INPUT * * *
 1900 WRITE(6, 9160)
```

```
9160  FORMAT(1H0, 10X, 'LIST OF DISTANCE'/)
      WRITE(6, 9180)(X(I), I=1, NENT)
9180  FORMAT(1H0, 10X, 10F10.3)
      READ(5, 9190)  VRED, TMIN
9190  FORMAT(2F10.3)
      IF(VRED.EQ.0.)GO TO 1930
      GO TO 1950
1930  VRED=VMAX
1950  IF(TMIN.NE.0.)GO TO 1970
      TMIN=X(NENT)   *(1./VMAX-1./VRED)+TI
1970  WRITE(6, 9195)  VRED, TMIN
9195  FORMTA(////, 11X, 'REDUCED VELOCITY
         VRED=', F10.3, 'KM/S'//, 11X,
         'THE LEAST REDUCED TIME TMIN=',F10.3,
         'SEC')
C     FINDING    GAMA    NGLE
C     ***GAM1***GAM3***GAM4***GAM2***
C     ***NG1***  NG3***  NG4***  NG2***
      READ(5, 9200)DGAM, IDFEN, C1, C2, FU,FWIL,
         FWIR, FO
9200  FORMAT(F8.5, I4, 6F8.5)
      IF(IDFEN.EQ.1)  IDFEN=2
      IF(FWIL.EQ.0)FWIL=FU
      IF(FWIR.EQ.0.)FWIR=FO
      A1=A(M)
      IF(M.EQ.1) GO TO 2005
      A1=AMAX(A, M)
2005  IF(C2.LE.A1) C2=A1+0.002
2008  IF(C1)  2010, 2010, 2020
2010  C1=9999.999
      GAM1=ARSIN(A(M)/C1)/GABO
      GO TO 2030
```

```
2020  GAM1=ARSIN(A(M)/C1)/GABO
2030  NG1=1
      GAMCR=ARSIN(A(M)/VMAX)/GABO
      GAM2=ARSIN(A(M)/C2)/GABO
      DCRAM=DGAM
      DGCR=0.
      NCRAM=1
      FNCR=0.
      IF(ISS(18).GT.0) WRITE(6,9208) I  SS(18),IDFEN
9208  FORMAT(11X, 'COS* *', I2, 'APPLY TO ANGLE
         WINDOW', I5, '*DGAM DEGREE'//)
      IF(ISS(19).GT.0) WRITE(6, 8209) ISS(19),FU,FW-
         IL, FWIR, FO
8209  FORMAT(11X, 'COS* *', I2, 'FREQUENCY
         WINDOW DOMAIN FU=', F5.2, 'FWIL=
         ', F5.2, 'FWIR=', F5.2, 'FO=', F5.2, HZ'//)
      IF(GAM1-GAMCR)  3906, 3905, 3905
3905  NG3=1
      GAM3=GAM1
      NG4=1
      GAM4=GAM1
      GO TO 2914
3906  IF(GAMCR-GAM1-1.) 2906, 2906, 2908
2906  GAM1=GAMCR   -   DGCR
      C1=A(M)/SIN(GAM1*GABO)
2908  NG3=(GAMCR-GAM1)/DGAM+1.0
      N=NG3
      IF(N/2-(N+1)/2) 2912, 2910, 2910
2910  NG3=NG3-1
      GAM1=GAM1+DGAM
2912  GAM3=(FLOAT(NG3)-1.)*DGAM+GAM1
      NG4=NG3+NCRAM-1
```

```
      GAM4=GAM3+(FLOAT(NCRAM)-1.)*DCRAM
      IF(GAM2-GAM4-1.) 2913, 2913, 2914
2913  GAM4=GAM2
      NG4=(GAM2-GAM3)/DCRAM+FLOAT(NG3)
2914  NG2=FLOAT(NG4)+(GAM2-GAM4)/DGAM
      N=NG2-NG4+1
      IF(N/2-(N+1)/2) 2920, 2918, 2918
2918  NG2=NG2-1
2920  CONTINUE
      GAM2=GAM4+(FLOAT(NG2)-FLOAT(NG4))*DG-
      AM C2=A(M)/SIN(GAM2*GABO)
      NGAM=NG2
      WRITE(6, 9210) C2, GAM2,C1, GAM1, VMAX, GA-
         MCR, DGAM, GAM1, GAM3, GAM4, GAM2,
         DCRAM, GAM3, GAM4, FU, FO
9210  FORMAT(//, 21X, 'MINIMUL PHASE VELOCITY
         C2=', F10.3,' KM/S    CORRESPO
      NDING ANGLE GAM2=',F10.3,'DEGREE'/21X,
         'MAXIMAL PHASE VELOCITY C1=',
         F10.3,'KM/S CORRESPONDING ANGLE
         GAM1=', F10.3, 'DEGREE'//21X, '
         MAXIMAL P-VELOCITY VMAX=',F10.3' KM/S
         CRITICAL ANGLE GAMCR=',
         F10.4, 'DEGREE'//21X, 'ANGLE INCREMENT
         DGAM=', F10.4,'DEGREE'/22X,
         'ANGLE WINDOW GAM1=', F10.3,'DEGREE'/
         35X,'GAM3=, F10.3,'DEGREE'/
         26X,'AND'/35X,'GAM4=', F10.3, 'DEGREE'/
         35X, 'GAM2=', F10.3, 'DEGREE'/
         26X, 'DCRAM=', F10.3,'DEGREE'/26X,'GAM3
         =', F10.3, 'DEGREE'/
         26X, 'GAM4=', F10.3, 'DEGREE'/26X, 'FU='
```

```
      F10.3/26X, 'FOO=', F10.3)
     WRITE(6, 8211) NG1, NG3, NG4, NG2
8211 FORMAT(1H0, 20X, 'NUMBER OF SAMPLE
      POINT'//35X, 'NG1=', I5, /35X, 'NG3=',
     I5/35X, 'NG4=', I5/35X, 'NG2=', I5/)
     IF(NGAM.LT.1000) GO TO 2048
     WRITE(6, 7211)
7211 FORMAT(1H0, 'NGAM.GT.1000')
     GOTO 6590
2048 IF(ISS(14).EQ.0) WRITE(6, 8212)
8212 FORMAT(1H0, 20X, 'SIMPSON-INTEGRATION'//)
     IF(ISS(14).EQ.1) WRITE(6, 8213)
C
8213 FORMAT(1H0, 20X, 'TRAPZ-INTEGRATION'//)
     IF(NGAM.LE.1000) GO TO 2921
     WRITE(6, 9211) NGAM
9211 FORMAT(1H0, 20X, 'NGAM=', I10, '.GT.1000')
     GO TO 6590
2921 CONTINUE
     GAM=GAM1-DGAM
     IF(NG3.EQ.1)  GAM=GAM1-DCRAM
     NWIND=IDFEN+1
     IF(ISS(24).EQ.0)  ISS(24)=1
     DISS=1./FLOAT(ISS(24))
     NFENS=NG2-IDFEN*ISS(24)
     IF(IDFEN.GT.0) DFEN=PI2/2./FLOAT(IDFEN)
     DO 2065 L=1, NGAM
     IF(L.GT.NG3.AND.L.LE.NG4) GO TO 2052
     GAM=GAM   +DGAM
     GO TO 2054
2052 GAM=GAM   +DCRAM
2054 SGAM=SIN(GABO*GAM)
```

```
      SQ=SGAM*SGAM
      SINA(L)=SGAM
      S1=0.
      A1=A(M)*A(M)
      DO 2063 J=1, M
      FF=2.
 2063 S1=S1+D(J)*SQRT(A1/(A(J)*A(J))-SQ)*FF
      COSA(L)=S1
      CALL SURFAC(A, B, RHO, M, ISS(17), SQ, GU,
         GV)
      FAC=SQRT(SGAM)*COS(GABO*GAM)
      IF(IDFEN.EQ.0.OR.L.LT.NFENS.AND.L.GT.NWI
         ND)  GO TO 2064
      IF(L.LE.NWIND) GO TO 2163
      FIF=L-NFENS
      FAC=FAC*(COS(DFEN*DISS*FIF)/2.+0.5)**
         ISS(18)
      GO TO 2064
 2163 FIF=L-1
      FAC=FAC*(0.5-COS(DFEN*FIF)/2.)**ISS(18)
 2064 DADA(L)=FAC*GV
      DIDI(L)=FAC*SGAM*GU
 2065 CONTINUE
C
      MHI=M-1
      NLAYER=NOL-MHI
      DO 2069 L=1, NLAYER
      LHI=L+MHI
      A  (L)=A(LHI)
      B  (L)=B(LHI)
      RHO(L)=RHO(LHI)
      D  (L)=D(LHI)
```

```
2069  CONTINUE
      ISIG=ISS(6)+1
      DO 6400 JSIG=1, ISIG
      READ(5, 9220) DT, NPTS, NA, N, T
9220  FORMAT(F10.3, 3I5, F10.3)
      WRITE(6, 9230)DT, NPTS, NA, N, T
9230  FORMAT(11X, 'SIGNAL PARAMETER'//, 11X,
         F15.3, 3I10, F15.3)
      IF(NPTS.LE.1024)  GO TO 2221
      WRITE(6, 7229)
7229  FORMAT(1H0, 'NPTS.GT.1024')
      GO  TO 6590
2221  IF(ISS(5).EQ.1)  GO TO 2500
      CALL SIGNAL(T, DT, N, V1, NPTS, NA, S)
      GO TO 2550
2500  ENDATA=-9999.
2550  FNPT=NPTS
      DAUER=DT*FNPT
      DO 2560 I=1, NPTS
2560  S(I)=S(I)/FNPT
C     ***DO FF   TTO   SIGNAL   SOURCE***
      CALL ZLOG(NPTS, MGA)
      CALL COOL(MGA, S, -1.0)
      INPT=NPTS/2
      IU=FU*DAUER  +1.0
      IF(IU.EQ.1) IU=2
      IO=FO*DAUER+1.0
      ILEFT=FWIL*DAUER+1.0
      IF(ILEFT.EQ.1) ILEFT=2
      IRIGHT=FWIR*DAUER+1.0
      NLEFT=ILEFT-IU
      NRIGHT=IO-IRIGHT
```

```
      IF(NLEFT.GT.0) DLEFT=PI2/2./FLOAT(NLEFT)
      IF(NRIGHT.GT.0) DRIGHT=PI2/2./FLOAT (NRI-
         GHT)
      DO 2710 I=1, INPT
      FIF=I-IRIGHT
      FAF=I-IU
      IF(I.GE.ILEFT.AND.I.LE.IRIGHT)  WAVEL(I)=S
         (I)
      IF(I.LE.IU.OR.I.GE.IO)  WAVEL(I)=(0., 0.)
      IF(I.GT.IU.AND.I.LT.ILEFT)
      WAVEL(I)=S(I)*(0.5-COS(DLEFT*FAF)/2.)**
         ISS(19)
      IF(I.GT.IRIGHT.AND.I.LT.IO)
      WAVEL(I)=S(I)*(COS(DRIGHT*FIF)/2.+0.5)
         **ISS(19)
 2710 CONTINUE
C
C     FINDING REFLECTION   COEFFICIENTS
C
      WRITE(6, 3399)
 3399 FORMAT(11X, 'FREQUENCY   IN   HZ'//)
      IM=(FU+FO)*DAUER/2.+1.0
      DO 3700 J=1, INPT
      FJ=J-1
      JZ=NPTS-J+1
      IF(J.LT.IU.OR.J.GT.IO)  GO TO 5150
      FREQ=FJ/DAUER
      WRITE(6, 3411) FREQ
 3411 FORMAT(11X, F10.3)

      DO 3600 I=1, NGAM
      U=SINA(I)/A  (1)
```

```
      CALL RECOPS(NLAYER, A, B, RHO, D, U, FR-
         EQ, CPP(I), CPS(I), CU, CV)
      PPR=REAL(CPP(I))
      PPI=AIMAG(CPP(I))
      CPP(I)=CMPLX(PPR, PPI)
      CPS(I)=CPP(I)
 3600 CONTINUE
C
      DO 6300 KE=1, NENT
      T0=X(KE)/VRED  +  TMIN
      CALL PRAEPI(NG1, NG2, NG3, NG4, KE, CU,
         CV, DGAM, DCRAM)
      W(KE, J)=CV
      W(KE, JZ)=CU
 6300 CONTINUE
      GO TO 5300
 5150 DO 5200 KE=1, NENT
      W(KE, J)=(0., 0.)
 5200 W(KE, JZ)=(0., 0.)
 5300 CONTINUE
 3700 CONTINUE
C     W(KE, 1)--W(KE,NPTS/2)   VERTICAL   COM
C        PONENT
C     W(KE, NPTS)--W(KE, NPTS/2+1)HORIZONTAL-
C     COMPONENT  (INVERSE ORDER)
      DO 5400 KE=1, NENT
      S(1)=(W(KE, 1)+EINI*W(KE, NPTS))*WAVEL(1)
      DO 5650 KF=2, INPT
      KF1=NPTS-KF+1
      KF2=KF1+1
      S(KF2)=(CONJG(W(KE, KF))+EINI*CONJG(W
         (KE, KF1)))*CONJG(WAVEL(KF))
```

```
5650 S(KF)=(W(KE, KF)+EINI* W(KE, KF1))* WAVEL
        (KF)
     INP1=INPT+1
     S(INP1)=(0., 0.)
C    * * *  FINDING INVERSE FFT   * * *
C    VERTICAL DISPLACEMENT OCCURSIN REAL
C    FIELD
C    HORIZONTAL DISPLACEMENT OCCURS IN IMA-
C    GEFIELD
     CALL COOL (MGA, S, +1.0)
     IF (ISS(4)-1) 6002, 6003, 6004
6002 L1=1
     L2=2
     GO TO 6008
6003 L1=1
     L2=1
     GO TO 6008
6004 L1=2
     L2=2
6008 DO 6299 KOM=L1, L2
     GO TO (6010, 6030), KOM
6010 DO 6020 J=1, NPTS
6020 BALLA(J)=REAL(S(J))
     GO TO 6050
6030 DO 6040 J=1, NPTS
6040 BALLA(J)=AIMAG(S(J))
6050 BALMAX=AMAX(BALLA, NPTS)
     BAL=ABS(AMIN(BALLA, NPTS))
     IF(BAL.GT.BALMAX) BALMAX=BAL
     IF(BALMAX.GT.AMP) AMP=BALMAX
     GO TO (6172, 6174), KOM
6172 WRITE(6, 9500) X(KE)
```

```
 9500  FORMAT(///, 11X, 'SEISMOGRAM(VERTICAL
          DISPLACEMENT)', F10.3, 'KM')
       GO TO 6176
 6174  WRITE(6, 9502) X(KE)
 9502  FORMAT(///, 11X, 'SEISMOGRAM(HORIZONTAL
          DISPLACEMENT)', F10.3, 'KM')
 6176  WRITE(6, 9510) NPTS, BALMAX
 9510  FORMAT (5X, 'NPTS=', I10, 5X,
          'BALMAX=', FI3.10)
       WRITE(6, 45)(BALLA(KC), KC=1, NPTS)
   45  FORMAT(10F13.5)
       IF(ISS(12).EQ.0)
          WRITE(16)(BALLA(K), K=1, NPTS)
 6299  CONTINUE
 5400  CONTINUE
 6400  CONTINUE
       CALL      FIG(AMP)
       GO TO 6600
 6590  WRITE(6, 9300)
 9300  FORMAT(1X, 'STOP DUE TO DATA ERROR'//)
 6600  CONTINUE
       STOP
       END
       CFOR, IN    SUB1
C      SUB1
       SUBROUTINE PRAEPI(NG1, NG2, NG3, NG4,
          KE, DU, DV, DGAM, DCRAM)
C      INTEGRATING BY NUMERICAL METHOD
       COMPLEX CU, CV, DU, DV
       DU=(0.0, 0.0)
       DV=(0.0, 0.0)
       IF(NG3.EQ.1) GO TO 10
```

```
      CALL INTGRA(KE, CE, CV, NG3, NG4, DCRAM)
      DU  =  DU  +  CU
      DV  =  DV  +  CV
      IF(NG3.EQ.NG4) GO TO 15
   10 CALL INTGRA(KE, CU, CV, NG3, NG4, DCRAM
      DU  =  DU  +  CU
      DV  =  DV  +  CV
   15 IF(NG4.EQ.NG2) GO TO 20
      CALL INTGRA (KE, CU, CV, NG4, NG2, DGAM)
      DU  =  DU+CU
      DV  =  DV+CV
   20 RETURN
      END
      CFOR, IN    SUB2
C     SUB2
      SUBROUTINE INTGRA (KE, CU, CV, NZ1, NZ2,
         EGAM)
      COMMON S(512), CPP(800), CPS(800), Z(100),
         D(100), A(100),
         B(100), RHO(100), X(30), SINA(800), COSA
         (800), PI2, DT, DAUER,
         FREQ, DGAM, C1, C2, PPR, PPI, TI, TO,
         ISS(30), INPT, NG1, NG2, NOL,
         NPTS, NENT, NGAM, DADA(800),DIDI(800),M
      COMPLEX S, CPP, CPS
      COMPLEX CU, CV, E
C     INTEGRATION FROM SIMPSON(ISS(14)=0) OR
C     TRAPEZIUM(ISS(14)=1)
      IF(ISS(14)) 10, 10, 20
C     SIMPSON
   10 WEIGHT  =  4.
      WTOT  =  3.
      GO TO 30
```

```
C
C      TRAPEZIUM
   20  WEIGHT  =  2.
       WTOT  =  2.
   30  PKA1  =  PI2 * FREQ/A(1)
       IF (FREQ.LE.0.0) PKA1  =  0.
       CALL PHASE (E, KE, NZ1, PKA1)
       CV=CPP(NZ1) * E * DADA(NZ1)
       CU=CPS(NZ1) * E * DIDI(NZ1)
       CALL PHASE(E, KE, NZ2, PKA1)
       CV=CV+CPP(NZ2) * E * DADA(NZ2)
       CU=CU+CPS(NZ2)*E*DIDI(NZI)
       NG11=NZ1+1
       NG21=NZ2-1
       NG12=NZ1+2
       NG22=NZ2-2
       DO 100 I=NG11, NG21, 2
       CALL PHASE(E, KE, I, PKA1)
       CV=CV+CPP(I) * E*DADA(I) * WEIGHT
  100  CU=CU+    CPS(I) * E * DIDI(I) * WEIGHT
       DO 200 I=NG12, NG22, 2
       CALL PHASE (E, KE, I,     PKA1)
       CV=CV+2. * CPP(I) * E * DADA(I)
  200  CU=CU+2. * CPS(I) * E * DIDI(I)
       ARGU=PKA1/(PI2 * X(KE))
       IF(ISS(5).EQ.0) ARGU=PKA1* PKA1* ARGU
       IF(ISS(5).EQ.1) ARGU=ARGU/(A(1) * A(1))
       AB  =  ABS(ARGU)
       BC  =  PI2 * FREQ * T0
       IF(ISS(5).EQ.1) BC=BC-PI2/4.
       E  =  CMPLX(0.0, BC)
       FAC  =  EGAM * 3.14156/180. * SQRT(AB)/WTOT
       CV  =  FAC * CV * CEXP(E)
```

```
      CU = -FAC * CU * CEXP(E)
      RETURN
      END
      CFOR, IN    SUB3
C     SUB3
      SUBROUTINE PHASE(E, KE, N, PA1)
C     COMPUTATION OF THE BESSEL FUNCTION
      COMMON S(512), CPP(800), CPS(800), Z(100),
         D(100), A(100),
         B(100), RH0(100),X(30),SINA(800), COSA(800),
         PI2, DT, DAUER,
         FREQ, DGAM, C1, C2, PPR, PPI, TI, T0, ISS
         (30), INPT, NG1, NG2, NOL,
         NPTS,NENT,NGAM,DADA(800), DIDI(800), M
      COMPLEX S, CPP, CPS
      COMPLEX E, EE
      IF(ISS(1).EQ.1) GOTO 100
      PH  =  -PA1*(X(KE)*SINA(N)+COSA(N))+0.7854
      EE  =  CMPLX(0. , PH)
      E=CEXP(EE)
      GO  TO 200
  100 PH  =  -PA1*COSA(N)
      EE  =  CMPLX(0., PH)
      E   = 2.*COS(PA1*X(KE)*SINA(N)-0.7854)*
         CEXP(EE)
  200 RETURN
      END
C     SUB4
      SUBROUTINE SURFAC(A, B, RHO, M, IS, SQ,
         GU, GV)
C
```

```
      EFFECT   OF   THE   TRANSMISSION   AND
C        THE  FREE  SURFACE
C
      DIMENSION A(1), B(1), RHO(1)
      A1  =  A(M)*A(M)
      U=1.0
      IF(M.EQ.1) GO TO 250
      DO 200 I  =  2, M
      J  =  I-1
      A2  =  RHO  (J)
      A3  =  RHO(I)
      C  =  A2*A3
      B1  =  A1/(B(J)*B(J))
      B2  =  A1/(B(I)*B(I))
      B3  =  SQRT(B1-SQ)
      B4  =  SQRT(B2-SQ)
      B5  =  SQRT(A1/(A(J)*A(J))-SQ)
      B6  =  SQRT(A1/(A(I)*A(I))-SQ)
      D1  =  2.*(A2/B1-A3/B2)
      D2  =  D1*SQ
      E  =  D2-A2
      F  =  D2+A3
      H  =  E+A3
      S  =  (E*B4-F*B3)/(SQ*H*H+B6*B4*E*E+
         C*B6*B3
         +B5*(D1*D2*B6*B3*B4+B3*F*F+C*B4))
      S=4.0*C*B5*B6*S*S
      U  =  U*S
  200 CONTINUE
  250 IF(IS.EQ.1) GO TO 300
      B1  =  A1/(B(1)*B(1))
      B2  =  SQRT(B1-SQ)
```

```
      B3  =  SQRT(A1/(A(1) * A(1)) - SQ)
      C  =  2. * SQ - B1
      D  =  - B1/2.
      Y  =  4.    * D/(C * C + 4. * B2 * B3 * SQ)
      GV  =  C * Y
      GU  =  - 2. * B2 * Y
      GO TO 400
300   GV  =  1.
      GU  =  1./SQRT(A1/(A(1) * A(1)) - SQ)
400   GV  =  GV * U
      GU  =  GU * U
      RETURN
      END
C     CFOR, IN   SUB5
C     SUB5
      SUBROUTINE RCR(D, A, NOL, TI, VMAX)
      DIMENSION D(1), A(1)
15    CALL MAXSN(NOL, A, VMAX, MAX)
      TI=0.0
      XCR=0.0
      M1=MAX-1
      IF(M1.EQ.0) GO TO 30
      DO 20 I=1, M1
      WINK=ARSIN(A(I)/A(MAX))
      FF=2.
      XCR=XCR+FF * D(I) * TAN(WINK
20    TI=TI+FF * D(I)/(COS(WINK) * A(I))
30    WRITE(6, 9000) TI, VMAX, XCR
9000  FORMAT(/////, 11X, 'TRAVEL TIME OF THE
        HEADWAVE'//, 26X,
         'T='F10.3, '+R/'F6.3, 'SEC'//, 11X, 'CRITIC
         ALDISTANCE
```

```
     ='F10.3,  'KM'//)
      RETURN
      END
      FOR, IN   SUB6
C     SUB6
      SUBROUTINE RECOPS (N, A, B, RHO, D, U,
         FREQ, RPP, RPS, RSS, RSP)
C
C     COMPUTATION OF P-SV REFLECTION
C     COEFFICIENTS
C
C     N=NUMBER OF DIFFERENT MEDIA, STARTING
C        ON TOP
C     A(I), B(I), RHO(I), (I=1, N)=P-VELOCITY,
C        S-VELOCITY AND DENSITY
      D(I), (I=2, N-1)=LAYER THICKNESS
C     U=PHASE SLOWNESS, FREQ=FREQUENCY
C     RPP, RPS, RSS, RSP=COMPLEX PP, PS, SS,
C        SP-REFLECTION COEFFICIENTS
      DIMENSION A(N), B(N), RHO(N), D(N)
      COMPLEX T1, T2, T3, T4, T5, RPP, RPS, RSS,
         RSP, DET, CN, CNS, T53, T63
         A, T11, T21, T31, T51, T61, T12, T15, T32, T45,
         T42, T35, T62, T65, T13, T23, T33
      PI=3.14159265
      OMEG=2.*PI*FREQ
      C=1./U
      RK=OMEG*U
      N1=N-1
      COM=C*OMEG
      U2=U*U
      C2=C*C
```

```
      RK2 = RK * RK
      OM2 = OMEG * OMEG
C     SET MATRIX ELEMENTS OF EQUATION (6)
      S = B(N)
      P = A(N)
      RRO = RHO(N)
      S2 = S * S
      P2 = P * P
      ARGP = 1. - C2/P2
      ARGS = 1.-C2/S2
      IF(ARGP.GE.0.) CN = CMPLX(0., - RK * SQRT
         (ARGP))
      IF(ARGP.LT.0.) CN = CMPLX(RK * SQRT
         (-ARGP), 0.)
      IF(ARGS.LT.0.) CNS = CMPLX(RK * SQRT
         (-ARGS), 0.)
      IF(ARGS.GE.0.) CNS = CMPLX(0., - RK * SQRT
         (ARGS))
      RL = 2.*RK2 - OM2/S2
      RPP = CN * CNS
      T1 = CMPLX(-S2 * S2 * RRO/(OM2 + OM2), 0.) *
         (CMPLX(4. * RK2, 0.) * RPP +
         CMPLX(RL * RL, 0.))
      T2 = CMPLX(0., 0.5) * CN
      T3 = CMPLX(0., - S2 * U/(2. * OMEG)) * (CMPLX
         (RL, 0.) + RPP + RPP)
      T4 = CMPLX(0., -0.5) * CNS
      T5 = CMPLX(-1./(2. * RRO * OM2), 0.) * (RPP
         + CMPLX(RK2, 0.))
      TR1 = REAL(T1)
      TI1 = AIMAG(T1)
      TR2 = REAL(T2)
```

```
      TI2=AIMAG(T2)
      TR3=2.*REAL(T3)
      TI3=2.*AIMAG(T3)
      TR4=REAL(T4)
      TI4=AIMAG(T4)
      TR5=REAL(T5)
      TI5=AIMAG(T5)
      IF(N.LT.3) GOTO 2000
C     SET MATRIX ELEMENTS (7)
C     DO MATRIX MULTIPLICATION (5) FROM LEFT
      TO RIGHT
C     DO NORMALIZATION
      DO 1000 J=2, N1
      I=N-J+1
      S=B(I)
      S2=S*S
      P=A(I)
      P2=P*P
      THK=RK*D(I)
      ARGP=1.-C2/P2
      IF(ARGP.GE.0.) GOTO 190
      RA=SQRT(-ARGP)
      P=THK*RA
      SP=SIN(P)
      CP=COS(P)
      X=RA*SP
180   ARGS=1.-C2/S2
      IF(ARGS.GE.0.) GOTO 200
      RB=SQRT(-ARGS)
      Q=THK*RB
      SQ=SIN(Q)
      CQ=COS(Q)
```

```
      Z=SQ*RB
      GO  TO 210
190   RA=-SQRT(ARGP)
      EP=0.5*EXP(THK*RA)
      IF(EP.EQ.0.) EP=1.
      EM=0.25/EP
      SP=EP-EM
      CP=EP+EM
      X=-SP*RA
      GO  TO 180
200   RB=-SQRT(ARGS)
      EP=0.5*EXP(THK*RB)
      IF(EP.EQ.0.) EP=1.
      EM=0.25/EP
      SQ=EP-EM
      CQ=EP+EM
      Z=-SQ*RB
210   W=SP/RA
      Y=SQ/RB
      G1=-2.*S2*U2
      G2=G1+1.
      E1=CP*CQ
      E2=1.-E1
      E3=W*Y
      E4=X*Z
      E5=W*CQ
      E6=Y*CP
      R1=COM*RHO(I)
      R2=1./R1
      R3=R1*G1
      R4=R1*G2
      F1=E2+E3
```

```
F2=F1 * R2
G16=-R2 * (F2+(E2+E4) * R2)
G13=-R3 * G16+F2
F3=G1 * F1+E3
F4=R3 * G13+F3
G31=R3 * F4+F3 * R4
G11=E1-F4
G33=F4+0.5
G61=-R3 * G31-R4 * (E3 * R4+F3 * R3)
G15=-R2 * (E5+Z * CP)
G23=-R3 * G15+E5
G21=-R3 * G23-R4 * E5
G12=R2 * (E6+X * CQ)
G32=-R3 * G12-E6
G51=-R3 * G32+R4 * E6
G22=E1
G25=Z * W
G52=X * Y
TR11=TR1 * G11+TR2 * G21-TI3 * G31+TR4 * G51+
    TR5 * G61
TI11=TI1 * G11+TI2 * G21+TR3 * G31+TI4 * G51+
    TI5 * G61
TR22=TR1 * G12 * TR2 * G22-TI3 * G32+TR4 * G52+
    TR5 * G51
TI22=TI1 * G12+TI2 * G22+TR3 * G32+TI4 * G52+
    TI5 * G51
TR33=-TI1 * G13-TI2 * G23+TR3 * G33-TI4 * G32-
    TI5 * G31
TI33=TR1 * G13+TR2 * G23+TI3 * G33+TR4 * G32+
    TR5 * G31
TR44=TR1 * G15+TR2 * G25-TI3 * G23+TR4 * G22
    +TR5 * G21
```

```
TI44=TI1*G15+TI2*G25+TR3*G23+TI4*  G22+
    TI5*G21
TR5=TR1*G16+TR2*G15-TI3*G13+TR4*G12+
    TR5*G11
TI5=TI1*G16+TI2*G15+TR3*G13+TI4*G12+TI5
    *G11
TR1=TR11
TI1=TI11
TR2=TR22
TI2=TI22
TR3=2.*TR33
TI3=2.*TI33
TR4=TR44
TI4=TI44
RMAX=ABS(TR5)
IF(RMAX.LT.ABS(TI5))  RMAX=TI5
IF(RMAX.LT.ABS(TI4))  RMAX=TI4
IF(RMAX.LT.ABS(TI3))  RMAX=TI3
IF(RMAX.LT.ABS(TI2))  RMAX=TI2
IF(RMAX.LT.ABS(TI1))  RMAX=TI1
IF(RMAX.LT.ABS(TR4))  RMAX=TR4
IF=RMAX.LT.ABS(TR3))  RMAX=TR3
IF(RMAX.LT.ABS(TR2))  RMAX=TR2
IF(RMAX.LT.ABS(TR1))  RMAX=TR1
IF(RMAX.EQ.0.)  RMAX=1.
RMAX=1./RMAX
TR1=TR1*RMAX
TR2=TR2*RMAX
TR3=TR3*RMAX
TR4=TR4*RMAX
TR5=TR5*RMAX
TI1=TI1*RMAX
```

```
      TI2=TI2*RMAX
      TI3=TI3*RMAX
      TI4=TI4*RMAX
      TI5=TI5*RMAX
 1000 CONTINUE
 2000 CONTINUE
C     SET MATRIX ELEMENTS  (8)
      P=A(1)
      P2=P*P
      S=B(1)
      S2=S*S
      RRO=RHO(1)
      ARGS=1.-C2/S2
      ARGP=1.-C2/P2
      IF(ARGP.GE.0.)  CN=CMPLX(0.,  -RK*SQRT
         (ARGP))
      IF(ARGP.LT.0.)  CN=CMPLX(RK*SQRT(-ARG-
         P), 0.))
      IF(ARGS.LT.0.)  CNS=CMPLX(RK*SQRT(-ARG-
         S), 0.)
      IF(ARGS.GE.0.)CNS=CMPLX(0.,-RK*SQRT(AR_
         GS))
      RM=RRO*S2
      RL=RK2+RK2-OM2/S2
      RPP=CN*CNS
      RM2=RM*RM
      RL2=RL*RL
      T11=CMPLX(-RK2, 0.)
      T13=T11+RPP
      T11=T11-RPP
      T21=CMPLX(0., RRO*OM2)
      T51=T21*CN
```

```
      T21 = - T21 * CNS
      T31 = CMPLX(0., - RM * RK * RL)
      RSS = CMPLX(0., 2. * RM * RK)*RPP
      T33 = T31 + RSS
      T31 = T31 - RSS
      T61 = CMPLX(- RM2 * RL2, 0.)
      RSS = CMPLX(4. * RK2 * RM2, 0.) * RPP
      T63 = T61 + RSS
      T61 = T61 - RSS
      T23 = CMPLX(0., RM * (2. * RK2 - RL))
      T53 = T23 * CN
      T23 = T23*CNS
      T12 = CMPLX(RK + RK, 0.)
      T15 = T12 * CNS
      T12 = T12 * CN
      T32 = CMPLX(0., 4. * RM * RK2)
      T45 = T32 * CNS
      T32 = T32 * CN
      T42 = CMPLX(0., 2. * RM * RL)
      T35 = T42 * CNS
      T42 = T42 * CN
      T62 = CMPLX(4. * RM2 * RL * RK, 0.)
      T65 = T62 * CNS
      T62 = T62 * CN
      T1 = CMPLX(TR1, TI1)
      T2 = CMPLX(TR2, TI2)
      T3 = CMPLX(TR3, TI3)
      T4 = CMPLX(TR4, TI4)
      T5 = CMPLX(TR5, TI5)
C     DO LAST PART OF MATRIX MULTIPLICATIO-
C         N(5)
C     COMPUTE REFLECTION COEFFICIENTS (4)
```

```
      DET=T1*T11+T2*T21+T3*T31+T4*T51+T5*T61
      IF(DET.EQ.0.)  DET  =1.
      DET=CMPLX(1., 0.)/DET
      RSS=T1*T13+T2*T23+T3*T33+T4*T53+T5*T63
      RSS=-RSS*DET
      RPP=-T1*T13-T2*T21-T3*T33+T4*T51-T5*T63
      RPP=RPP*DET
      T3=T3*CMPLX(0.5, 0.)
      RPS=T1*T12+T3*T32+T3*T42+T5*T62
      RPS=-RPS*DET
      RSP=T1*T15+T3*T35+T3*T45+T5*T65
      RSP=RSP*DET
      RETURN
      END
C     CFOR, IN   SUB 7
C     SUB7
      SUBROUTINE SIGNAL(T, DT, N, V1, NPTS,
     NA, S)
C     COMPUTATION  OF  THE  SIGNAL
      DIMENSION S(1)
      COMPLEX S
      DO 100 I=1, NA
100   S(I) = (0.0, 0.0)
      NS = T/DT + 1.
      FN = N
      FM = (FN + 2.)/FN
      FMQ = 1./(FM*FM)
      D = FN*3.14159265/T
      TT=-DT
      A=0.0
      DO 200 I=1, NS
      TT=TT+DT
```

```
      T1=D*TT
      T2=FM*T1
      T3=3.36*T1
      C=(1.-COS(T1)+FMQ*(COS(T2)-1.))/D
      S(NA+I)=CMPLX((C, 0.0)
      C=ABS(SIN(T1)-SIN(T2)/FM)
      IF(C.GT.A) A = C
  200 CONTINUE
      A = V1/A
      DO 300 I = 1, NS
      J = NA + I
  300 S(J) = A*S(J)
      J = NA + NS + 1
      A=3.14159265/FLOAT(NPTS-J+1)
      DO 400 I = J, NPTS
  400 S(I)=0.5*(1.+COS(FLOAT(I-J+1)*A))*S(NS)
      RETURN
      END
C     CFOR, IN  SUB 8
C     SUB8
      SUBROUTINE COOL(N, X, SIGN)
C     DO FFT
      DIMENSION M(25)
      DIMENSION X(2)
      COMPLEX X, HOLD, WK, Q, V, DI
      DI=(0.0, 1.0)
      LX=2**N
      DO 1 I=1, N
    1 M(I)=2**(N-I)
      DO 4 L=1, N
      NBLOCK=2**(L-1)
      LBLOCK=LX/NBLOCK
```

```
      LBHALF=LBLOCK/2
      K=0
      DO 4 IBLOCK=1, NBLOCK
      FK=FLOAT(K)
      FLX=FLOAT(LX)
      V=6.2831853*SIGN*FK/FLX*DI
      WK=CEXP(V)
      ISTART=LBLOCK*(IBLOCK-1)
      DO 2 I=1, LBHALF
      J =ISTART+I
      JH=J +LBHALF
      Q=X(JH) *WK
      X(JH) = X(J) -Q
      X(J) = X(J) +Q
    2 CONTINUE
      DO 3 I=2, N
      II=I
      IF(K.LT.M(I))  GO TO 4
    3 K=K-M(I)
    4 K=K+M(II)
      K=0
      DO 7 J =1, LX
      IF (K.LT.J)  GO TO 5
      HOLD=X(J)
      X(J)=X(K+1)
      X(K+1)=HOLD
    5 DO 6 I=1, N
      II=I
      IF  (K.LT.M(I))  GO TO 7
    6 K=K-M(I)
    7 K=K+M(II)
      RETURN
```

```
      END
CFOR, IN SUB9
C     SUB9
      SUBROUTINE ZLOG(N, M)
      DO 10 I=1, 1, 12
      IF(N-2**I)10, 5, 10
    5 M=I
      GO TO 20
   10 CONTINUE
   20 RETURN
      END
C     CFOR, IN SUB 10
C     SUB 10
      FUNCTION AMAX(A, N)
      DIMENSION A(1)
      AMAX=A(1)
      DO 10 I=2, N
      IF(A(I)-AMAX)  10, 10, 9
    9 AMAX=A(I)
   10 CONTINUE
      RETURN
      END
      CFOR, IN SUB11
C     SUB11
      FUNCTION AMIN(A, N)
      DIMENSION A(1)
      AMIN=A(1)
      DO 10 I=2, N
      IF(A(I)-AMIN)  9, 10, 10
    9 AMIN=A(I)
   10 CONTINUE
      RETURN
```

```
      END
CFOR, IN      SUB 12
C     SUB 12
      SUBROUTINE MAXSN(LX, X, XMAX, INDEX)
      DIMENSION X(1)
      XMAX=X(1)
      DO 10 I=1, LX
   10 XMAX=AMAX1(XMAX, X(I))
      DO 20 J=1, LX
      INDEX=J
      IF(X(J)-XMAX)20, 30, 20
   20 CONTINUE
   30 RETURN
      END
```

# 第十章 任意介质中的射线追踪方法

前面我们讨论了常速水平层、常速倾斜层、速度随深度线性变化的非均匀层走时的正演计算和反演计算。但实际的地壳和上地幔界面常是弯曲的，介质的速度分布往往也是非均匀的。如何计算它们的走时（正演问题），又如何根据实际观测走时推断地壳和上地幔界面的起伏情况及介质的速度分布（反演问题），就是本章讨论的中心问题。

射线追踪是研究任意介质分布地震波传播问题的有效方法。原则上它可以用几何作图的方法来作，但由于费时且精度差而失去其实用价值。因此，实际上射线追踪一般是计算机来完成的。本章的重点是讨论计算机进行射线追踪的具体计算方法和计算机程序，有关射线追踪的原理仅作简要叙述。

近年来，对射线追踪正演问题国内外都进行了大量的研究。Stein(1969) 发表了在恒定的垂直速度梯度情况下，利用二维速度分布的一种计算机程序，速度不同的地区用一个多边形隔开。Will(1975) 引入了一种归纳法，这样速度梯度能够有随机方向。Dantz(1978) 提出了一种程序，其二维模式由恒速度线确定，具有恒速度和恒梯度的地区以及由倾斜线构成的速度分布地区，调不同的子程序进行计算。Gebrande (1975) 用解析函数，例如反正切、逼近速度等值线，较清晰地表示了复杂地质构造的波速情况，如忱状构造或消减带一类比较复杂的速度模型。Pšenčik(1975) 等许多人使用了一种不同但更普通的方法，他们用一阶常微分方程描述射线路径，空间坐标 $x_i$ 和 $z_j$ 的矩形网格决定出局部速度 $v(x_i, z_i)$，在相邻四个点的长方形中使用了特殊的非线性插值函数。Cerveny等人使用了样条函数，走时和射

线踪迹用 Runge-Kutta 法计算。

国内，国家地震局地球所、成都地质学院、国家地震局郑州物探大队等单位对上述方法都作了一定程度的研究和改进。

射线追踪反演问题的论文相对说来发表要少些。成都地质学院和国家地震局郑州物探大队等单位用阻尼最小二乘法、广义逆等方法进行了射线追踪反演计算，编写了相应的计算机程序。

## §10-1 射线追踪原理简述

射线追踪就是把在介质中连续传播的波离散成一条条射线来研究，从震源出发将射线分成若干段，逐段求出射线的轨迹和走时，依次累加，最后得出地下介质中射线分布和地面上的时距曲线。

众所周知，在介质速度变化不太大，波长比较短的情况下，波的传播可以用波前方程来描述：

$$\left(\frac{\partial w}{\partial x}\right)^2+\left(\frac{\partial w}{\partial y}\right)^2+\left(\frac{\partial w}{\partial z}\right)^2=n^2. \qquad (10\text{-}1.1)$$

式中 $n$ 是折射率，$w$ 是波阵面(波前)。

波前方程(10-1.1)是一阶偏微分方程，它的解只是空间坐标的函数。具有非常简单的形式，

$$w(x,\ y,\ z)=\text{常数}.$$

它表示了三维空间的曲面簇，也就是波的传播可以视为从震源散开的一个个空间曲面。每一时刻对应一个曲面，这个曲面就是该时刻的波阵面，波阵面的法线就是所谓的射线，射线的方向就是波的传播方向。波强度的各种变化可以用射线的收缩与扩张来描述，更重要的是到达的时间可以由沿射线路径的积分得到。

从波前方程很容易看出波阵面的形状与介质的速度分布有关。在均匀介质中速度是常数，相同时间间隔的波阵面之间对应

点的距离相等，即对应点的曲率不变，因此法线的方向也不会变，也就是说均匀介质中的射线是直线。在非均匀介质中，速度分布是空间坐标的函数，在相同时间间隔内波阵面前进的距离不一样，波阵面的曲率发生了变化，波阵面的法线方向也跟着发生变化，也就是说非均匀介质中的射线是曲线。

如果已知速度分布和边界条件，可以求出波前方程的解析解或数值解。也可以把非均匀介质中弯曲的射线，用若干个圆弧来逼近。即把非均匀介质分成若干薄层，把每个薄层的速度视为是线性变化的。众所周知，速度线性变化的介质，地震射线的轨迹是圆弧。因此很容易分别求出射线在各个薄层的轨迹和走时，依次累加，最后得到射线在非均匀介质中的射线分布和走时。

## §10-2 射线追踪正演计算

本节结合后面的射线追踪程序，将详细地叙述射线追踪的具体计算方法和步骤。由于均匀介质和非均匀介质中射线轨迹和走时计算公式不一样，为了便于理解，我们将分别叙述。这并不是说程序也是分开编写的，相反，为了使程序简洁，节省内存，缩短编译时间，凡两者相同部份均合二为一。射线追踪计算总要先有界面和速度模型，因此本节首先叙述这个问题。

### （一）界面及速度模型的描述

描写界面及速度模型的方法很多，下面我们介绍几种常用的方法。

(1) 用反正切多项式描述界面。反正切函数 $z(x)=\mathrm{arctg}x$，当 $|x|$ 增加到一定程度后，其函数值变化很小；若干个反正切函数相加，只有其中某一个贡献大，而其它的贡献都很小；每一个反正切曲线可以用很少的参数确定其基本形状。利用反正切曲线的上述特性，可以比较方便地用下述具有若干个反正切函数的多

项式来描述各种复杂界面。

$$z_k(x)=a_{0k}+a_{1k}x+\sum_{i=1}^{n}\Delta Z_{ki}\left(\frac{1}{2}+\frac{1}{\pi}\text{arc}\,\text{ctg}\frac{x-x_{0ki}}{b_{ki}}\right),$$

$$(k=1,2\cdots,m). \quad (10\text{-}2.1)$$

式中，下角标$k$表示界面的序号；下标$i$表示第$k$个界的第$i$个反正切曲线的序号；$n$表示第$k$层反正切曲线的个数；$m$表示界面的个数（包括地面）；$\Delta Z_{ki}$表示反正切曲线的落差；$X_{0ki}$表示反正切曲线中心的横坐标；$b_{ki}$表示反正切曲线的陡度，即过反正切曲线$\frac{1}{4}\Delta Z_{ki}$处作反正切曲线顶线的平行线，中点到该平行线的距离。(10-2.1)式各参数的意义参看图10-2-1。

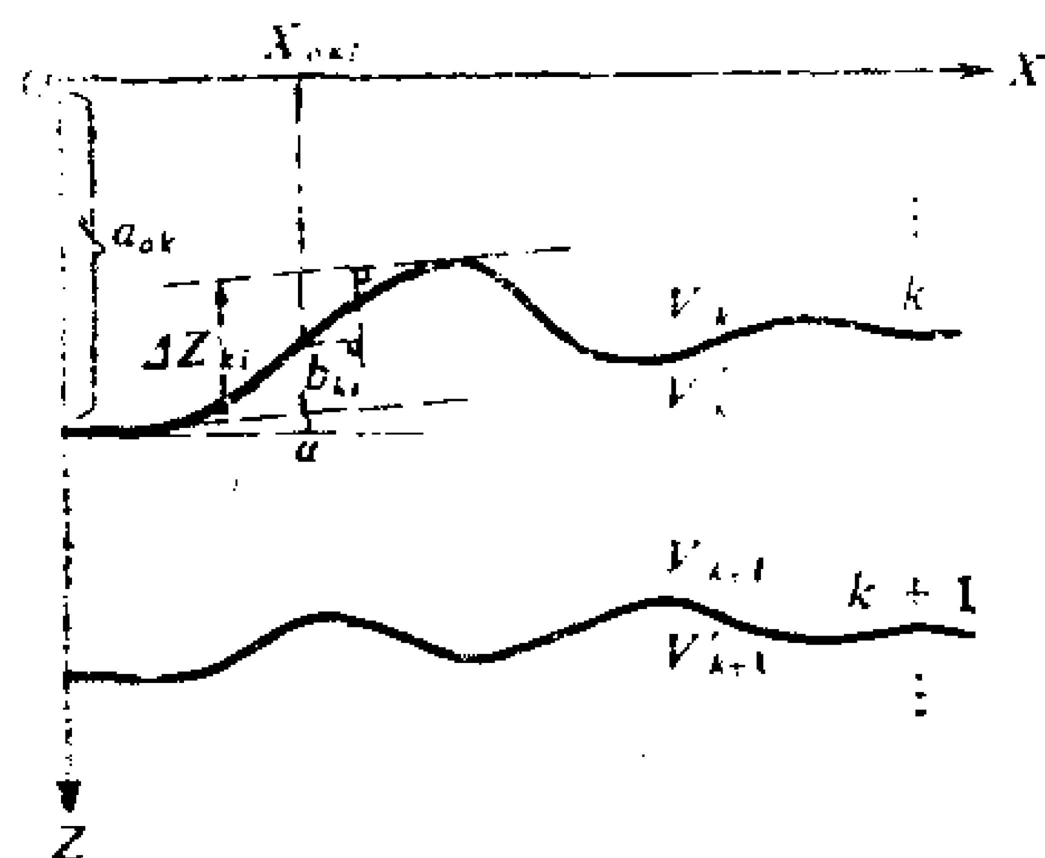



图10-2-1 用反正切曲线描述界面

如果第$k$个界面是水平界面，（则式(10-2-1)为

$$z_k(x)=a_{0k}.$$

如果第$k$个界面是倾斜界面，界面的斜率为$a_{1k}$则式(10-2.1)为

$$z_k(x)=a_{0k}+a_{1k}x.$$

(2) 给出界面一系列离散坐标($x_i$, $z_i$)值，然后用样条

曲线或抛物线把这些点连起来，如图10-2-2所示。

上述描述界面的方法各有其优缺点。第一种方法所用的参数少，省机时，适于射线追踪反演，但要精确地描述界面和人工确定其参数比较困难。第二种方法能够准确直观地确定其参数，比较精确地描述界面，但参数多，费机时，一般不宜用来反演。

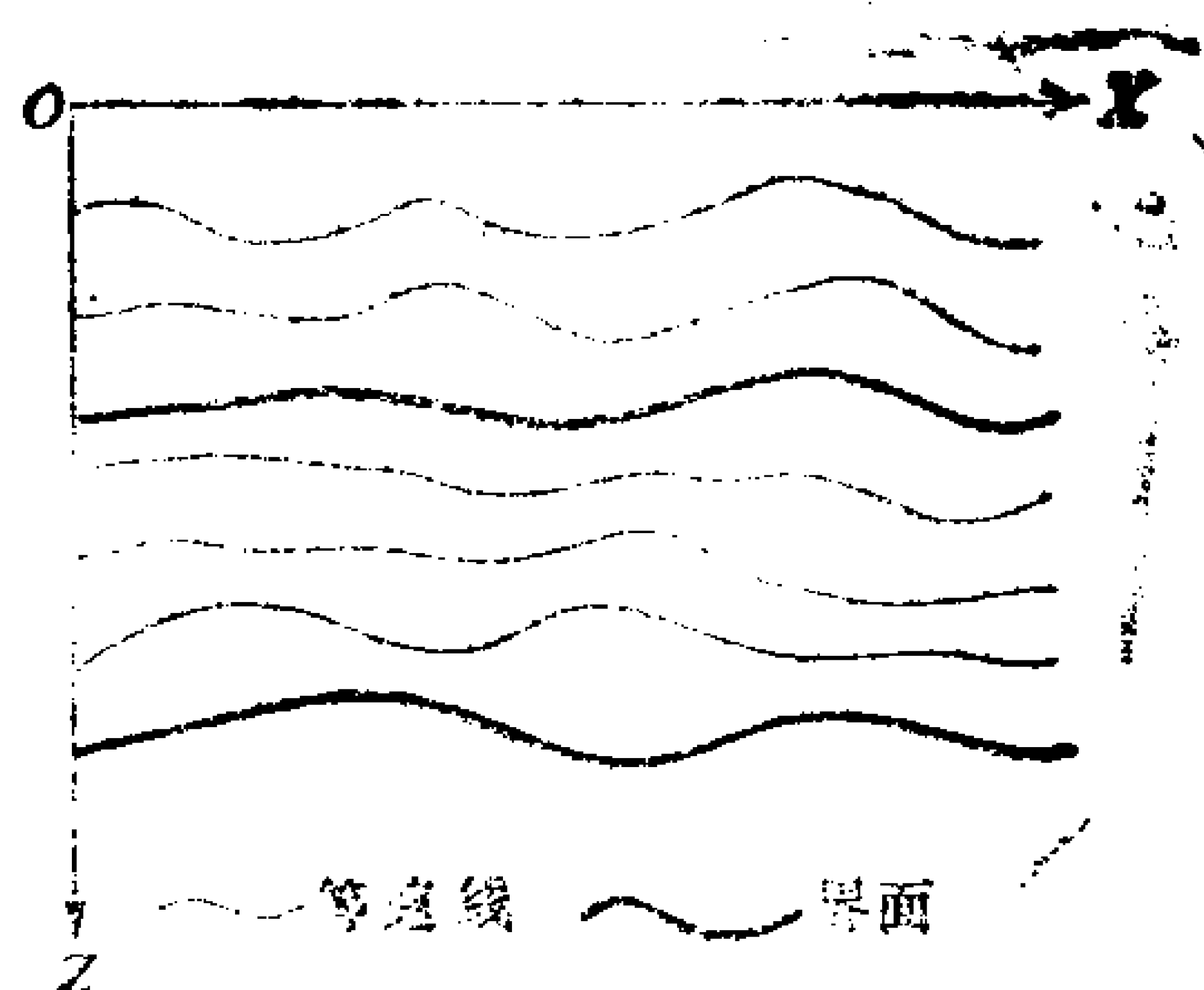



图10-2-2 用样条曲线连接离散点表示的界面和等速线

非均匀介质的速度模型一般用等速线表示，上述描述界面的方法都可以用来描述等速线。也可以给出非均匀介质的顶面速度 $V'_k$，底面速度 $V_{k+1}$，$V'_k \neq V_{k+1}$，如果顶底界面之一是弯曲界面，那么介质内任意一点的速度一般是不相等的（参看图10-2-1）。均匀介质的速度模型一般直接给出各层速度值。

## （二）射线追踪计算步骤

射线追踪就是从震源出发，按规定的步长逐步前进，逐段计算射线每前进一步的轨迹和走时，依次累加，直到射线达到或超过界面。如果射线超过界面，求射线与界面的交点。射线在界面要发生反射和折射，需要求出射线与界面交点处的界面倾角、反射角、折射角、临界角等。以射线与界的交点为起点重复上述步骤，直到射线达到目的界面。求出目的界面反射线与水平线的夹角，将此夹角视为初始角进行上行射线追踪直到地面为止。于是得到了一条经过地下介质到达地面的射线再改变震源处的初始角，重复上述步骤，最后得到了一条条经过地下介质到达地面的射线。将各条射线到达地面的走时和炮检距画成一条曲线即为所要求的时距曲线。下面分别叙述均匀和非均匀介质中射线追

踪的具体计算步骤。

**1．均匀介质中的射线追踪计算步骤**

(1) 计算目的层下界面第 $s$ 条射线在震源处与横轴的夹角 $\alpha_s$(以下称初始角)。为了使各条射线到达地面之间的距离大致相等，按下式计算初始角 $\alpha_s$

$$\alpha_s = \operatorname{arctg} \frac{a_{0k}}{s \cdot \Delta x}$$

式中 $a_{0k}$ 是坐标原点到目的层 k 下界面的直垂距离，$s$ 是第 $s$ 条射线，$\Delta x$ 是炮检距的一半。

(2) 从震源$O$出发，沿射线方向前进一个步长 $R$，到达 $A$ 点。计算 $A$ 点的横坐标 $X_A$，纵坐标$Z_A$，射线到达 $A$ 点的走时 $T_A$，由图10-2-3很容易得到：

$$X_A = R \cdot \cos\alpha_s,$$

$$Z_A = R \cdot \sin\alpha_s,$$

$$T_A = \frac{R}{V_1},$$

式中 $V_1$ 是第一层的速度。

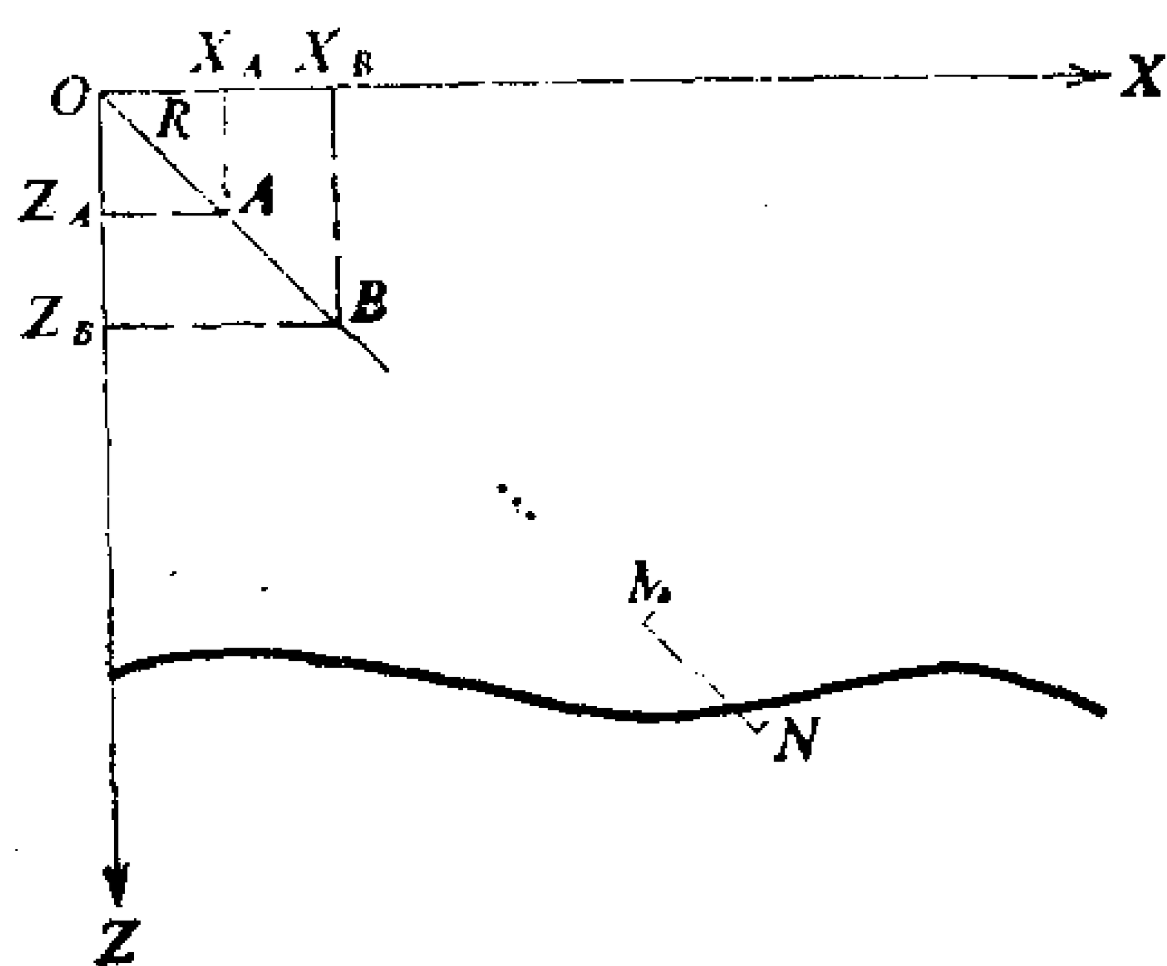


图 10-2-3 均匀介质中的射线追踪

(3) 判断 $A$点是否到达或超过第一层下界面。具体作法如下，求出横坐标为 $X_A$ 时第一层下界面的纵坐标 $Z(X_A)$，令 $\Delta Z_A = Z_A - Z(X_A)$，如 $|\Delta Z_A|$ 小于或等于某个小量 $\varepsilon_z$（例如0.01公里），表示 $A$ 点到达第一层下界面；如果 $\Delta Z_A > \varepsilon_z$表示 $A$ 点超过第一层下界面，如果 $\Delta Z_A < \varepsilon_z$ 表示 $A$ 点还没有到达第一层下界面，此时从 $A$ 点出发，继续以步长 $R$ 前进，到达 $B$ 点。如同射线从

$O$ 点到达 $A$ 点一样，计算 $B$ 点的横坐标$X_B$，纵坐标$Z_B$，到达 $B$ 点的走时 $T_B$，

$$X_B = X_A + R \cdot \cos\alpha_s,$$
$$Z_B = Z_A + R \cdot \sin\alpha_s,$$
$$T_B = T_A + \frac{R}{V_1}.$$

判断 $B$ 点是否达到或超过第一层下界面，如果仍然没有，重复上述计算，直到射线达到和超过界面为止。

（4）如果射线超过界面，用“二分法”求射线与界面的交点。设第 $m$ 次跨步，射线从 $M$ 点跨过界面到达 $N$ 点。求出 $MN$的中点 $F_1$，计算 $F_1$ 的横坐标 $X_{F1}$，纵坐标 $Z_{F1}$

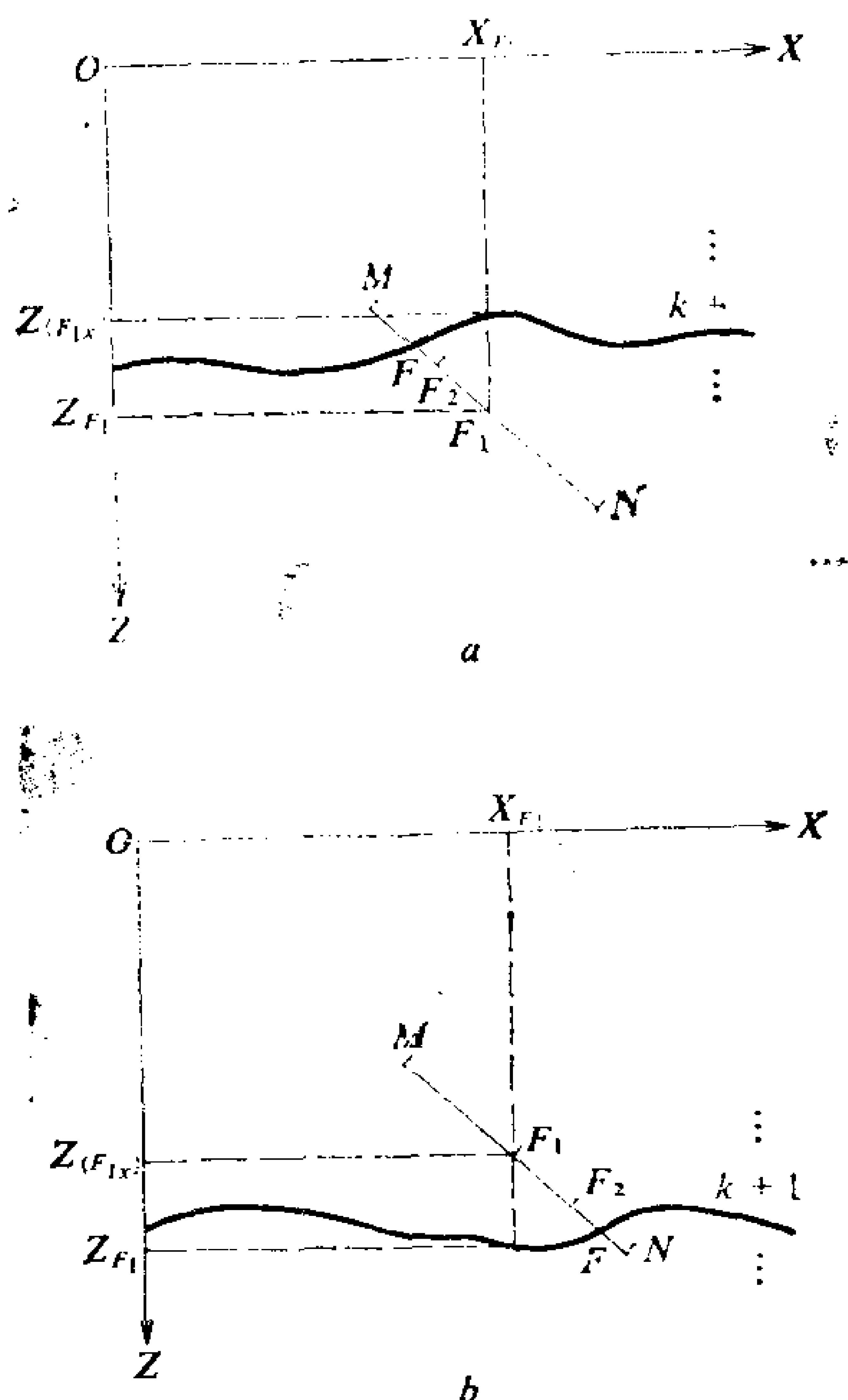



图 10-2-4 均匀介质中用“二分法”求射线与界面的交点

a.二分点超过界面； b.二分点未达到界面

$$X_{F1} = X_M + \frac{1}{2} R\cdot\cos\alpha_s,$$
$$Z_{F1} = Z_M + \frac{1}{2} R\cdot\sin\alpha_s.$$

式中$X_M$ 和$Z_M$ 分别是 $M$ 点的横坐标和纵坐标。如图10-2-4所示

计算横坐标为$X_{F1}$时界面的纵坐标$Z(X_{F1})$，令$\Delta Z_1 = Z_{F1} - Z(X_{F1})$，如果$|\Delta Z_1|$小于等于事先给定的精度$\varepsilon_z$，则认为$F_1$是射线与界面的交点。如果$|\Delta Z_1|$大于$\varepsilon_Z$则不能认为$F_1$是射线与界面的交点，继续进行“二分”。在继续进行“二分”之前，先要判断$F_1$是在界面的上方或者是在界面的下方。如果$F_1$在界面的下方，如图 10-2-4 a 所示，则新的“二分”点$F_2$的横坐标$X_{F2}$，纵坐标$Z_{F2}$分别为

$$X_{F2} = X_{F1} - \frac{1}{2^2} R\cdot\cos\alpha_s = X_{F1} - \frac{1}{4} R\cdot\cos\alpha_s,$$

$$Z_{F2} = Z_{F1} - \frac{1}{2^2} R\cdot\sin\alpha_s = Z_{F1} - \frac{1}{4} R\cdot\sin\alpha_s.$$

如果$F_1$在界面的上方，如图 10-2-4 b 所示，则

$$X_{F2} = X_{F1} + \frac{1}{2^2} R\cdot\cos\alpha_s = X_{F1} + \frac{1}{4} R\cdot\cos\alpha_s,$$

$$Z_{F2} = Z_{F1} + \frac{1}{2^2} R\cdot\sin\alpha_s = Z_{F1} + \frac{1}{4} R\cdot\sin\alpha_s.$$

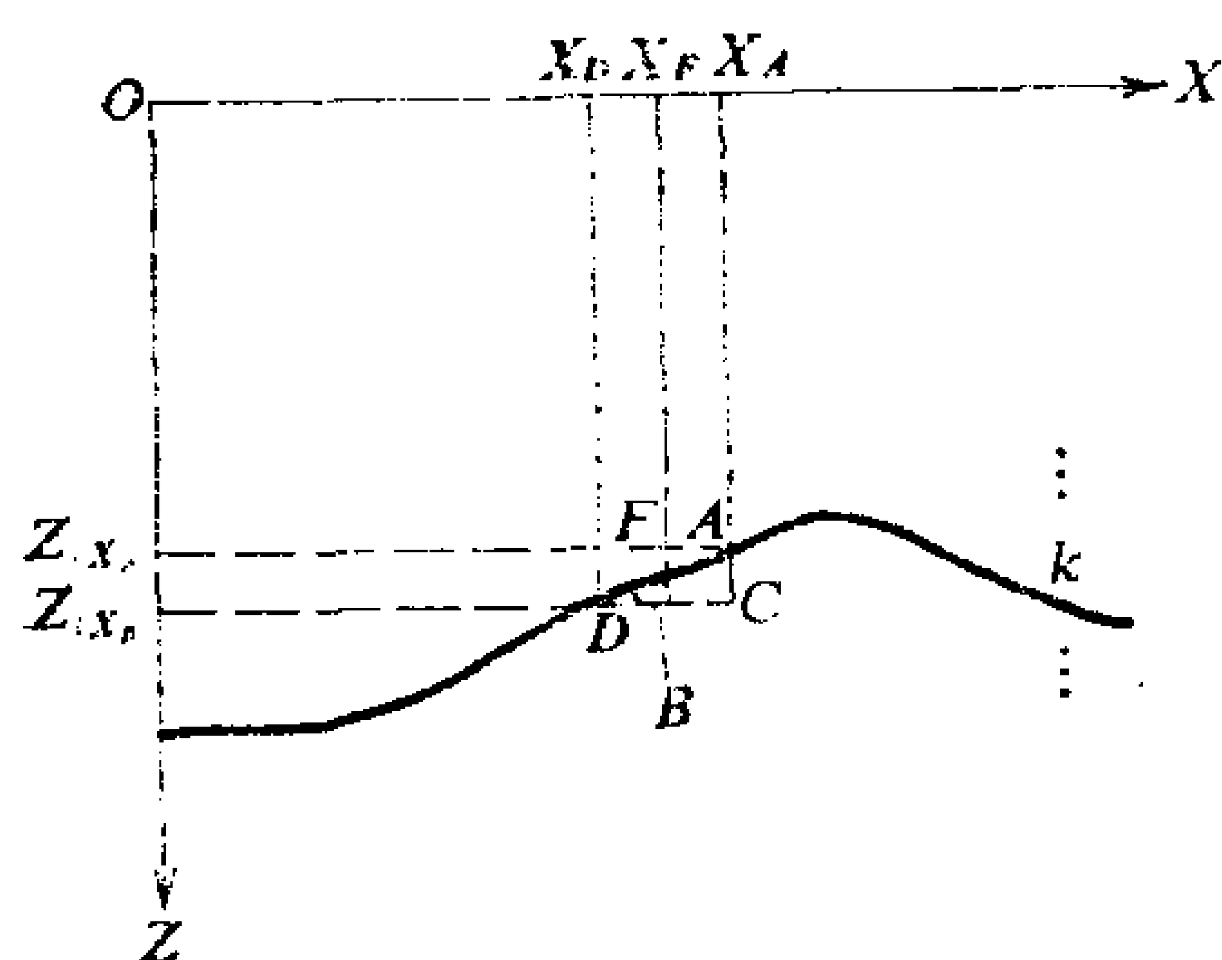


图10-2-5 计算界面倾角$B$

判断新的“二分”点是否是射线与界面的交点，如果仍然不是，继续“二分”，直到求出射线与界面的交点为止。

(5) 求射线与界面交点处的界面倾角$B$。

如图10-2-5所示，设射线与界面的交点为$F$，$F$点的横坐标

为 $X_F$。从 $F$ 点沿横轴方向前进 $\Delta X$ 得到 $A$ 点的横坐标 $X_A$，后退 $\Delta X$ 得到 $D$ 点的横坐标 $X_D$，

$$X_A = X_F + \Delta X,$$
$$X_D = X_F - \Delta X.$$

$\Delta X$ 一般取界面参数 $b_{ki}$ 最小者的十分之一。分别计算 $X_A$ 和 $X_D$ 对应的界面纵坐标 $Z(X_A)$ 和 $Z(X_D)$，则 $F$ 点的界面倾角

$$B = \mathrm{arctg}\left(\frac{Z(X_A) - Z(X_D)}{2 \cdot \Delta X}\right).$$

（6）如果下行射线到达的界面不是目的界面，计算下行入射线与法线的夹角

$$A_{11} = \frac{\pi}{2} - (E - B).$$

式中 $E$ 是射线与横轴的夹角。计算下行折射线与横轴的夹角

$$A_{12} = \arcsin(V'_k \cdot \sin A_{11} / V_k).$$

式中 $V_k$ 和 $V'_k$ 分别是界面 $k$ 上、下的速度。计算下行射线与横轴的夹角

$$B_{12} = \frac{\pi}{2} - (A_{12} - B),$$

计算下行射线的临界角

$$A_D = \arcsin(V_k / V'_k).$$

如果 $A_{11}$ 小于 $A_D$，将 $B_{12}$ 视为初始角进行下行射线追踪直到目的界面为止。

（7）射线到达目的界面后，计算反射线与水平线的夹角 $B_{11}$

$$B_{11} = E - B.$$

将 $B_{11}$ 视为初始角进行反射波上行射线追踪。如果上行射线碰到界面求上行射线与法线的夹角

$$A_{11}=\frac{\pi}{2}-B_{11}.$$

计算上行折射线与法线的夹角

$$A_{21}=\arcsin(V_k\cdot\sin A_{11}/V_k').$$

计算上行射线的临界角

$$A_u=\arcsin(V_k'/V_k).$$

计算射线与水平线的夹角

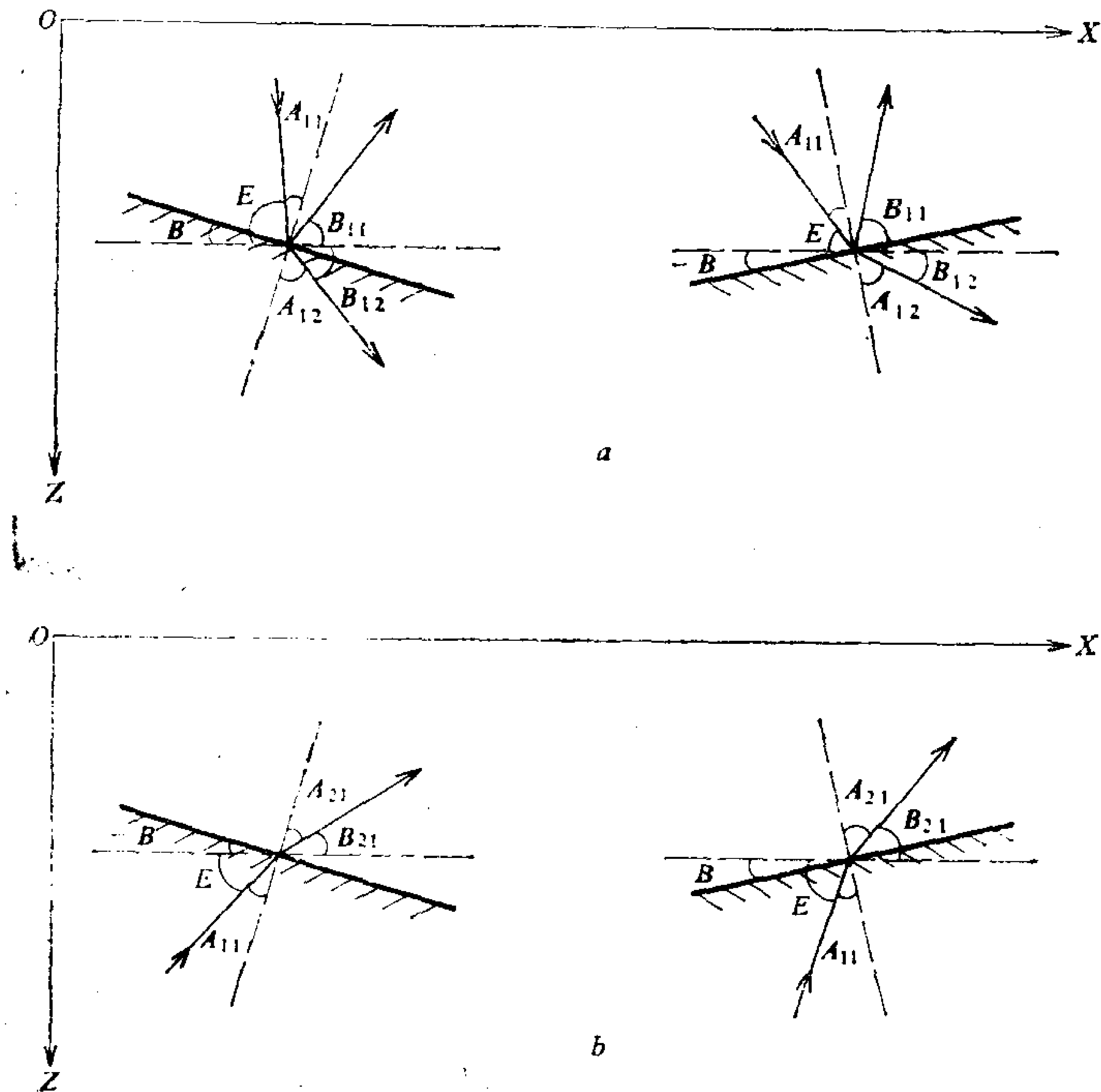



图 10-2-6 射线在界面各种角的关系

a.下行射线；　　b.上行射线

$$B_{21}=\frac{\pi}{2}-A_{21}+B.$$

如果 $A_{11}$ 小于 $A_u$ 将 $B_{21}$ 视为初始角进行上行射线追踪直到地面为止。以上各种角参看图10-2-6。

(8) 减小震源处的初始角，重复上述步骤，进行另一条小角度反射波射线追踪，直到入射线在目的界面的入射角等于或略大于临界角。

(9) 当入射角等于或略大于临界角时，进行首波射线追踪，如图10-2-7所示。在临界点 $G$ 以上行射线与横轴的夹角 $B_{11}$ 为初始角，进行一次上行射线追踪。从临界点 $G$ 出发，沿横轴方向前进一个步长 $\Delta S$（例如2公里），射线沿界面滑行到 $G_1$ 点。计算 $G_1$ 点的横坐标 $X_{G1}$，纵坐标 $Z_{G1}$，到达 $G_1$ 点的走时 $T_{G1}$，

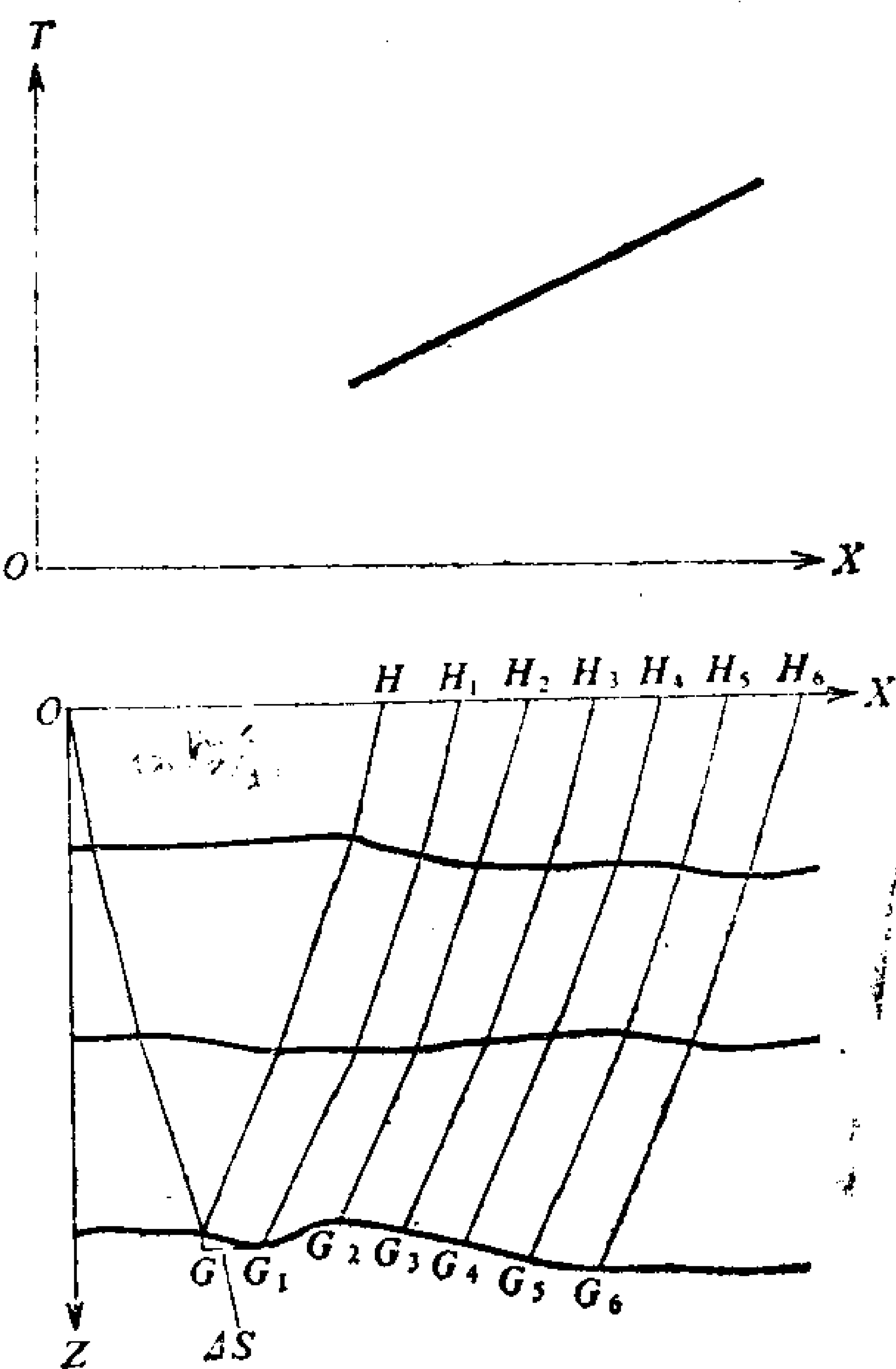



图10-2-7 均匀介质中折射波射线追踪
（上图走时曲线，下图射线图）

$$X_{G1}=X_G+\Delta S,$$
$$Z_{G1}=Z(X_{G1}),$$
$$T_{G1}=T_G+\frac{\sqrt{Z(X_G)^2+Z(X_{G1})^2}}{V_k'}.$$

式中 $X_G$ 是 $G$ 点的横坐标，$T_G$ 是下行射线到达临界点 $G$ 的走时。计算首波射线在 $G_1$ 点的起射角

$$A_{G1}=\frac{\pi}{2}-\left[A_D+\operatorname{arctg}\left(\frac{Z(X_{G1})-Z(X_G)}{\Delta S}\right)\right].$$

从 $G_1$ 点出发，将射线与横轴的夹角 $A_{G1}$ 视为初始角进行上行首波射线追踪，方法与小角度反射波上行射线追踪一样，设从 $G_1$ 出发的上行首波射线与地面的交点为 $H_1$，则炮检距为 $OH_1$ 的首波走时为

$$T_{H_1}=T_{G_1}+T'_{H_1}.$$

式中 $T'_{H1}$ 是从 $G_1$ 出发的上行首波射线的走时。

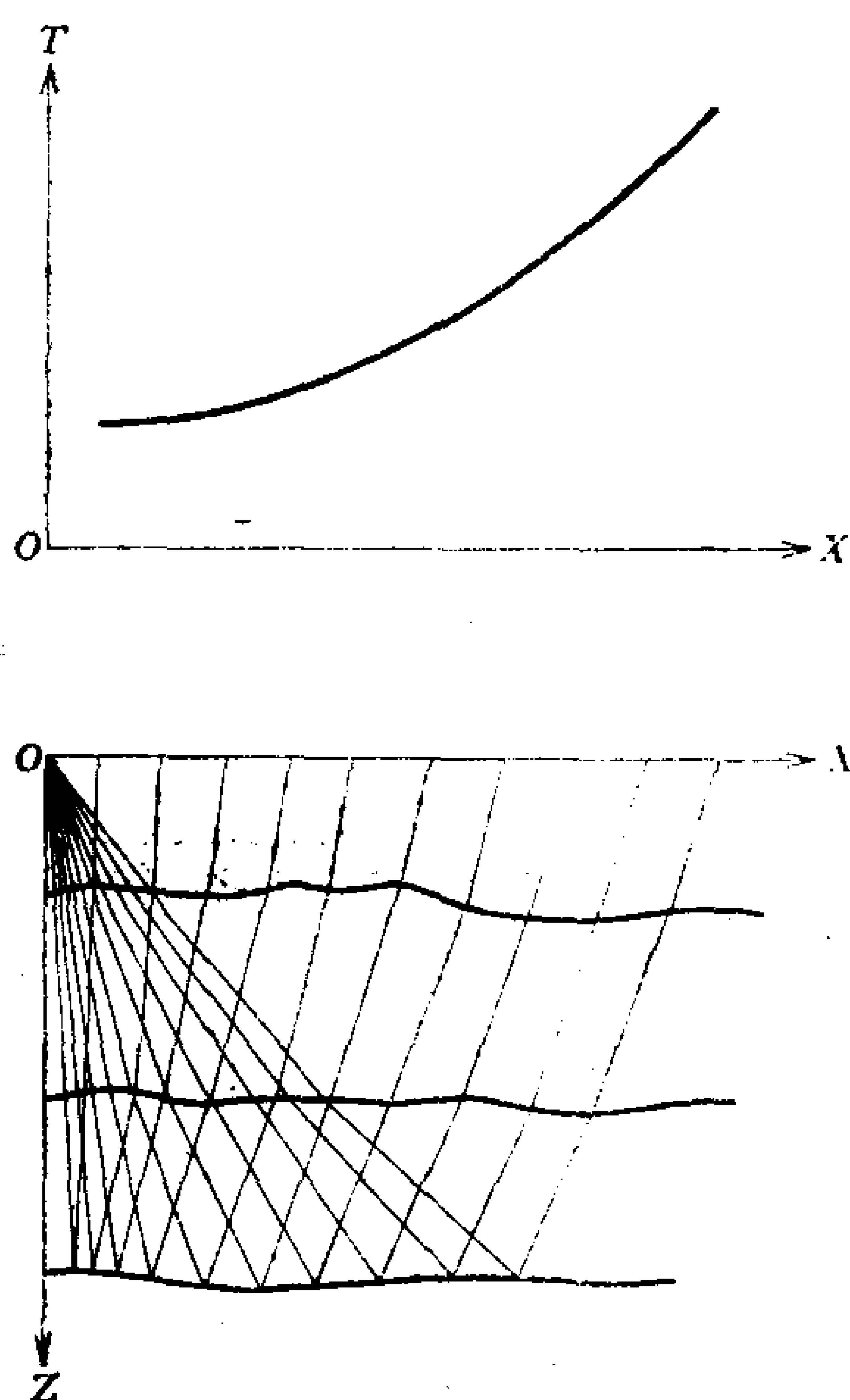


图 10-2-8 反射波射线追踪(上图走时曲线，下图射线图)

继续以 $\Delta S$ 步前进，每前进一步，重复上述步骤，直到计算完事先规定的最后一条射线为止。至此得到了目的层界面的首波射线图和时距图。

(9) 继续减小震源处的初始角，进行广角度反射波射线追踪。追踪方法同小角度反射波射线追踪。一直到计算完事先规定的最后一条射线为止。最后得一幅完整反射波射曲线图和时距曲线，如图10-2-8所示。

(10) 射线追踪正演计算一般是从第一层开始，依次往下作，直到作完规定的最后一层为止。

**2. 弯曲界面任意非均匀介质中的射线追踪计算步骤**

任意非均匀介质中的射线追踪，是把非均匀层分成若干细

层，把每个细层的速度看成是线性变化的，而各个细层的速度梯度不一定相等。细层的分法不是沿纵坐标轴或某个方向等，分而是以跨步起点地震射线曲率半径和曲率中心为基准前进一个角度步长 $Q$，如图10-2-10所示。求出每向前跨一步的射线坐标和走时，依次累加，最后得到剖面的射线分布和时距曲线。下面叙述其主要计算步骤。

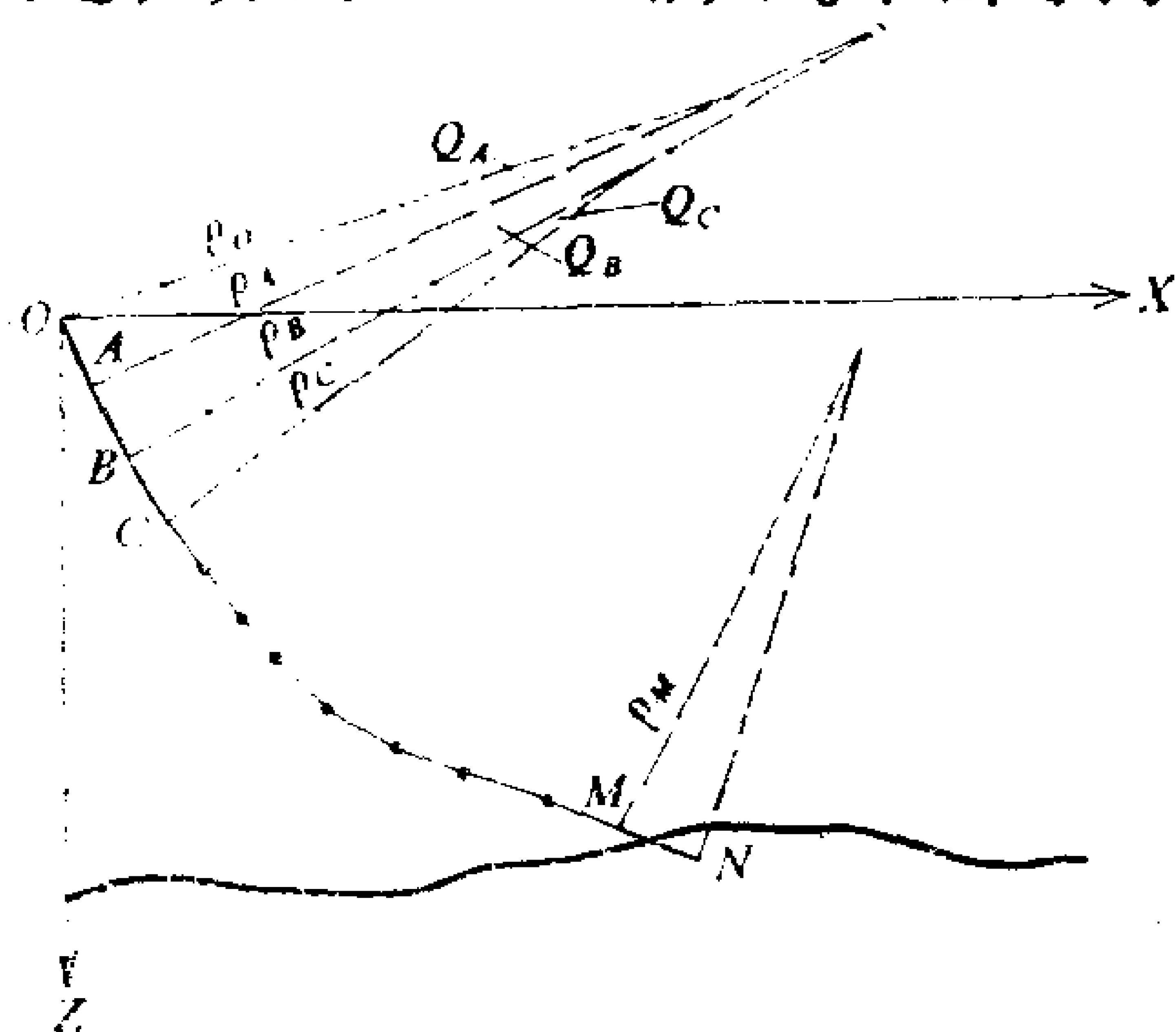



图 10-2-9 非均匀介质中的射线追踪

（1）计算第 $s$ 条射线在震源处的初始角 $\alpha_s$，方法与均匀介质相同。

（2）射线每向前跨一步都必须计算射线跨步起点的速度梯度 $G_n$，

$$G_n = \left|\sqrt{G_X^2 + G_Z^2}\right|.$$

式中 $G_X$ 和 $G_Z$ 分别是跨步起点的速度横向梯度和纵向梯度。对于给出非均匀层顶底面速度 $V_k'$ 和 $V_{k+1}$ 这种速度模型，$G_X$ 和 $G_Z$ 分别为

$$G_X = \Delta V_X / \Delta X,$$
$$G_Z = \Delta V_Z / \Delta Z.$$

式中 $\Delta V_X$ 是跨步起点沿横轴方向跨 $\Delta X$ 的速度增量，$\Delta V_Z$ 是非均匀层顶底界面的速度差，$\Delta Z$ 是跨步起点顶、底界面的距离。

计算第 $n$ 次跨步起点的速度 $V_n$。对于给出非均匀层顶、底面速度 $V_k'$、$V_{k+1}$ 的速度模型，

$$V_n = V_k' + \Delta V_Z \cdot \Delta H.$$

式中 $\Delta H$ 是跨步起点到顶界面的距离。

对于以速度横向梯度 $G_X$ 和纵向梯度 $G_Z$ 描述的速度模型

$$V_n = V_k' + \sqrt{(G_X \cdot S_X)^2 + (G_Z \cdot S_Z)^2}$$

式中 $S_X$ 和 $S_Z$ 分别是跨步起点距震源或非均匀层顶面射线出发点的水平距离和垂直距离。

计算第 $n$ 次跨步起点与横轴的夹角

$$\alpha_n = \alpha_s \pm (n-1)Q.$$

式中 $\alpha_s$ 是第 $s$ 条射线在震源处或界面出发点处射线与横轴的夹角，$Q$ 是角度步长，如果进行下行射线追踪 $\alpha_s$ 后面取正号；如果进行上行射线追踪 $\alpha_s$ 后面取负号。

计算第 $n$ 次跨步起点速度梯度方向与射线方向之间的夹角 $i_n$，

$$i_n = \mathrm{arctg}\left(\frac{\mathrm{tg}A_n - \mathrm{tg}\alpha_n}{1 + \mathrm{tg}A_n \cdot \mathrm{tg}\alpha_n}\right).$$

式中$A_n$是第$n$次跨步起点处速度梯度方向与横轴之间的夹角。

计算第 $n$ 次跨步起点的曲率半径

$$\rho_n = \frac{V_n}{G_n \cdot \sin i_n}.$$

计算第 $n$ 次跨步端点的横坐标 $X_{n+1}$，纵坐标 $Z_{n+1}$。对于下行射线

$$X_{n+1} = X_n + (\rho_n \cdot \sin Q) \cdot \cos\alpha_n + \rho_n \cdot (1 - \cos Q) \cdot \sin\alpha_n,$$
$$Z_{n+1} = Z_n + (\rho_n \cdot \sin Q) \cdot \sin\alpha_n + \rho_n \cdot (1 - \cos Q) \cdot \cos\alpha_n.$$

对于上行射线

$$X_{n+1} = X_n + (\rho_n \cdot \sin Q) \cdot \cos\alpha_n - \rho_n \cdot (1 - \cos Q) \cdot \sin\alpha_n,$$
$$Z_{n+1} = Z_n - (\rho_n \cdot \sin Q) \cdot \sin\alpha_n - \rho_n \cdot (1 - \cos Q) \cdot \cos\alpha_n,$$

计算第 $n$ 次跨步后射线到达端点的走时

$$T_{n+1}=T_n+\frac{1}{2G_n}\left[\ln\frac{1+\cos\alpha_n}{1-\cos\alpha_n}-\ln\frac{1+\cos(\alpha_n+Q)}{1-\cos(\alpha_n+Q)}\right].$$

如果射线方向与速度方向之间的夹角为零度，则按下式计算 $X_{n+1}$、$Z_{n+1}$ 和 $T_{n+1}$；

$$X_{n+1}=X_n+R\cdot\cos\alpha_n,$$
$$Z_{n+1}=Z_n\pm R\cdot\sin\alpha_n,$$
$$T_{n+1}=T_n+\frac{1}{G_n}\ln\frac{V_{n+1}}{V_n}.$$

式中 $R$ 的意义与均匀介质中的射线追踪相同，$V_n$ 是第 $n$ 次跨步起点的速度，$V_{n+1}$ 是端点的速度。上面各式中 $X_n$、$Z_n$ 和$T_n$ 分别是第 $n$ 次跨步起点的横坐标、纵坐标及走时。

每跨一步都必须判断射线达到和超过界面没有，如果没有继续跨步，直到射线达到或超过界面为止。

（3）如果射线超过界面，用“二分法”求射线与界面的交点。设第 $m$次跨步，射线从 $M$ 点跨过界面到达 $N$ 点，如图10-2-10 a和 b 所示。将角

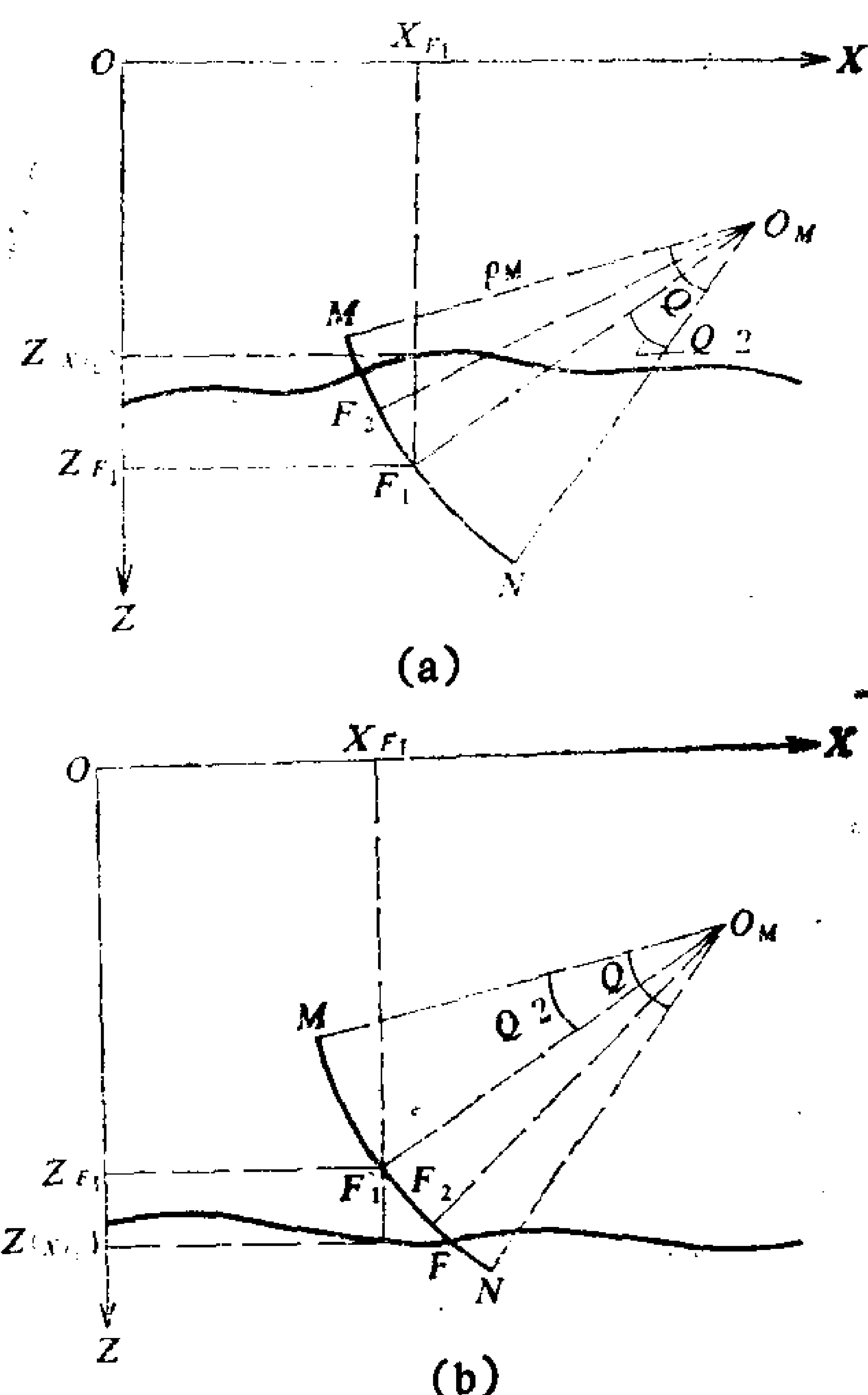



图10-2-10　非均匀介质中，用“二分法”求射线与界面的交点

a.二分点超过界面；b.二分点未达到界面

$Q$等二分，得到二分后的角度步长$Q_1$，

$$Q_1=\frac{1}{2}Q,$$

以$M$点的曲率中心为圆心，曲率半径为半径，前进角度步长$Q_1$，射线到达$F_1$。按上述方法计算$F_1$点在追踪坐标系中的横坐标$X_{F1}$，纵坐标$Z_{(F1)}$。判断$F_1$点到达界面没有，方法与均匀介质中的射线追踪相同。如果仍然超过界面，继续“二分”，在上次前进的二分角度步长$Q_1$中减去这一次二分角度步长$\frac{1}{2^2}Q$，得到新的角度步长，

$$Q_2=Q_1-\frac{1}{2^2}Q=\frac{1}{4}Q,$$

如图10-2-10a所示。如果没有达到界面，则在上次前进的二分角度步长$Q_1$中加上这一次的二分角度步长$\frac{1}{2^2}Q$，

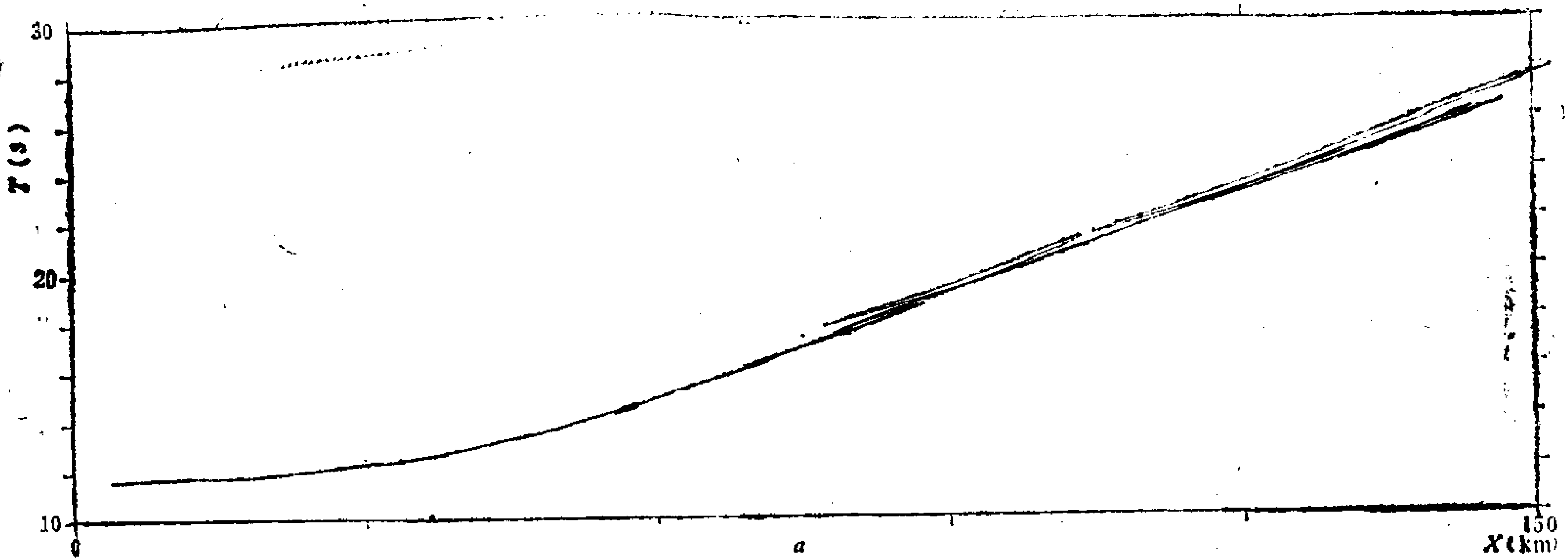
$$Q_2=Q_1+\frac{1}{2^2}Q=\frac{3}{4}Q,$$

如图 10-2-10 b所示。射线达到新的二分点$F_2$，判断$F_2$达到界面没有，如果没有，重复上述步骤，继续二分，直到射线达到界面为止。

以下的射线追踪步骤与均匀介质中射线追踪步骤（5）—（10）基本一样。主要区别是在进行下行射线追踪时，每跨一角度步长都必须判断下行射线与横轴的夹角是否小于1°。如果小于1°，进行回折波射线追踪，追踪方法与上行反射波射线追踪大致相同，唯一的差别仅在于回折点上行射线与横轴的夹角总是视为0°进行上行射线追踪，一直计算到事先规定的射线条数为止。

当计算完事先规定的最后一条射线后，我们就得到了目的界面的反射波和首波走时曲线，目的层的回折波走时曲线以及它们相应的射线图。

按照上述步骤，编写出射线追踪正演程序，可以迅速方便地



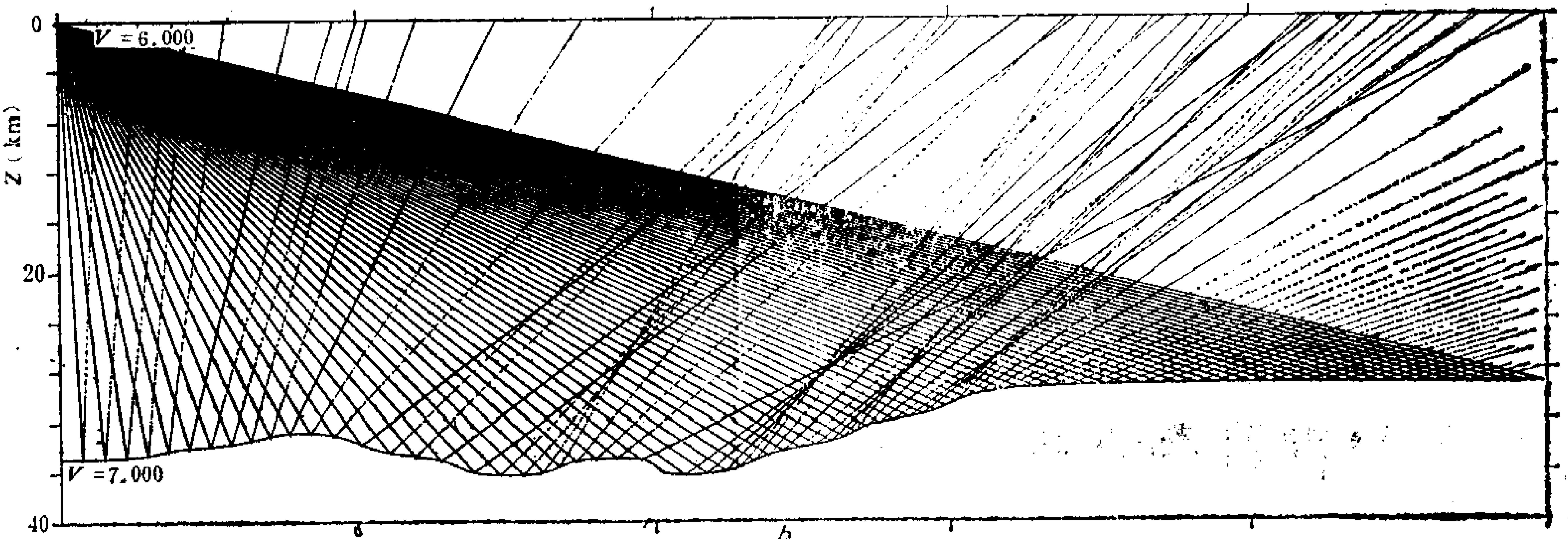




图 10-2-11 弯曲界面的时距曲线和射线分布图

a. 时距曲线图；  b. 射线分布图

计算和绘出水平、倾斜和各种弯曲界面的反射波、首波和回折波的走时曲线和射线分布图。下面举几个计算的典型例子。

图 10-2-11 表示了均匀介质中弯曲界面的时距曲线图和射线分布图。从射线图上可以看到地震射线在地下介质中的分布情况，传播的路径。与地震记录对比，可以发现地震射线在地表分布密集的地方振幅较大，反之振幅较小。界面曲率中心位于界面的下方，反射线超前，曲率中心位于界面的上方，反射线回转，界面曲率越大超前或回转越厉害，反射线分布不均匀。在射线发生超前回折的地方时距曲线也相应发生超前、回折，出现多值。

图 10-2-12 是多层平缓起伏界面的反射波时距曲线和射线分布图。从图中可以看出，下层界面的时距曲线和射线分布除了受反射界面的影响外还要受上面各层界面起伏情况和速度分布的影响。

图 10-2-13 是非均匀介质弯曲界面反射波和回折波射线分布图和时距曲线。从射线图中可以看出非均匀介质中的射线是曲线，曲率大小与该点的速度梯度成正比，与速度成反比，与射线方向及速度梯度方向之间夹角的正弦成正比。反射线除了受介质的速度分布影响之外，还要受界面形状的影响。

图 10-2-14 是非均匀介质弯曲界面的首波射线分布图和时距曲线图。首波射线也是曲线，影响射线曲率的因素与反射线相同，时距曲线的形状除了与介质速度分布有关外，还与界面的形状有关。

## §10-3 射线追踪反演计算

射线追踪反演计算就是根据地面测量的折射或反射走时，利用射线追踪方法去计算地下地震界面及速度分布有关参数。由于反演中计算理论走时不是用解析方法，而是一个追踪过程，因此我们称为射线追踪反演。

本章采用地球物理反演中比较有效的阻尼最小二乘法。并考

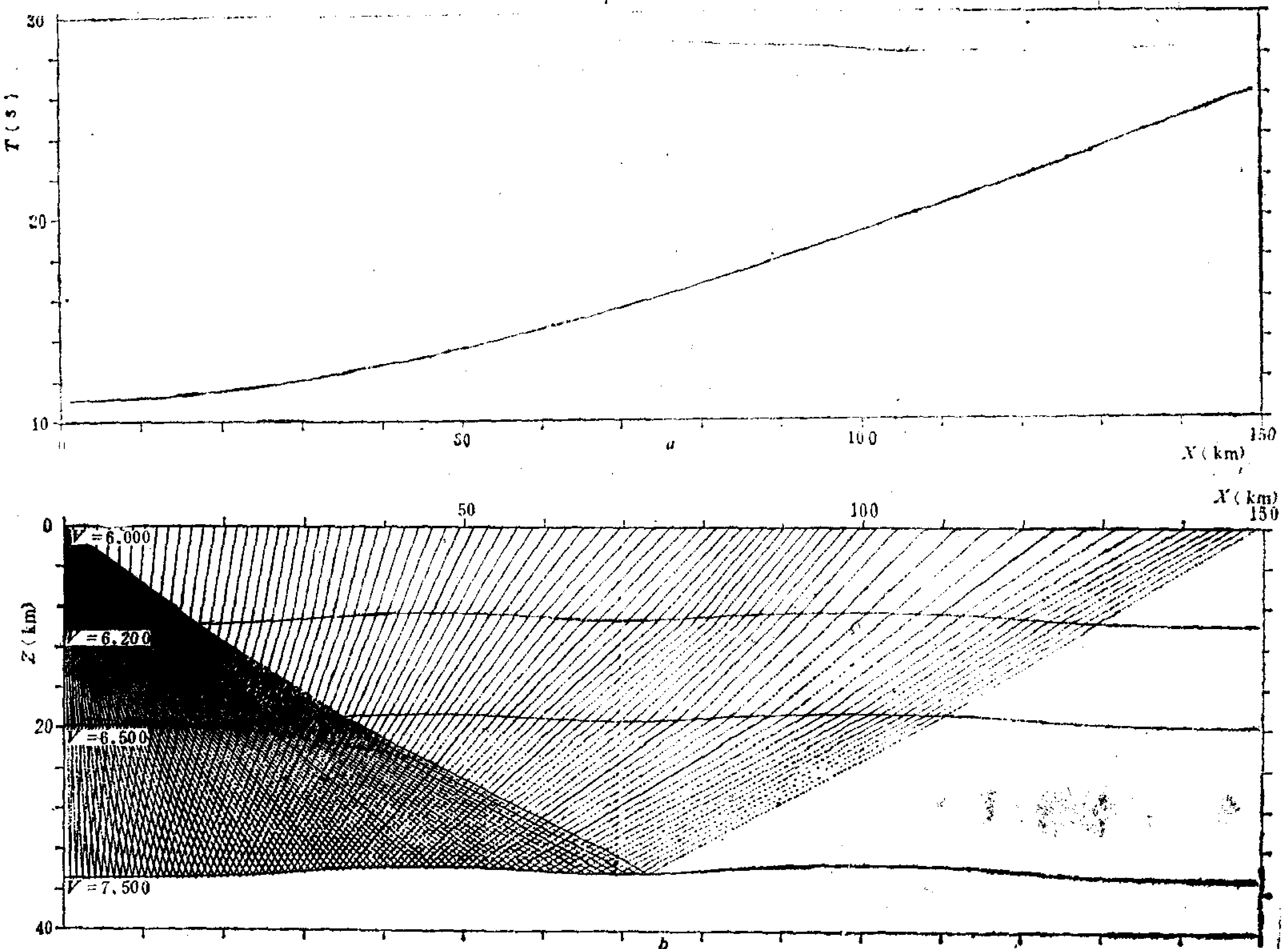



图 10-2-12 均匀介质中平缓起伏界面的时距曲线和射线分布图

a.时距曲线图； b.射线分布图

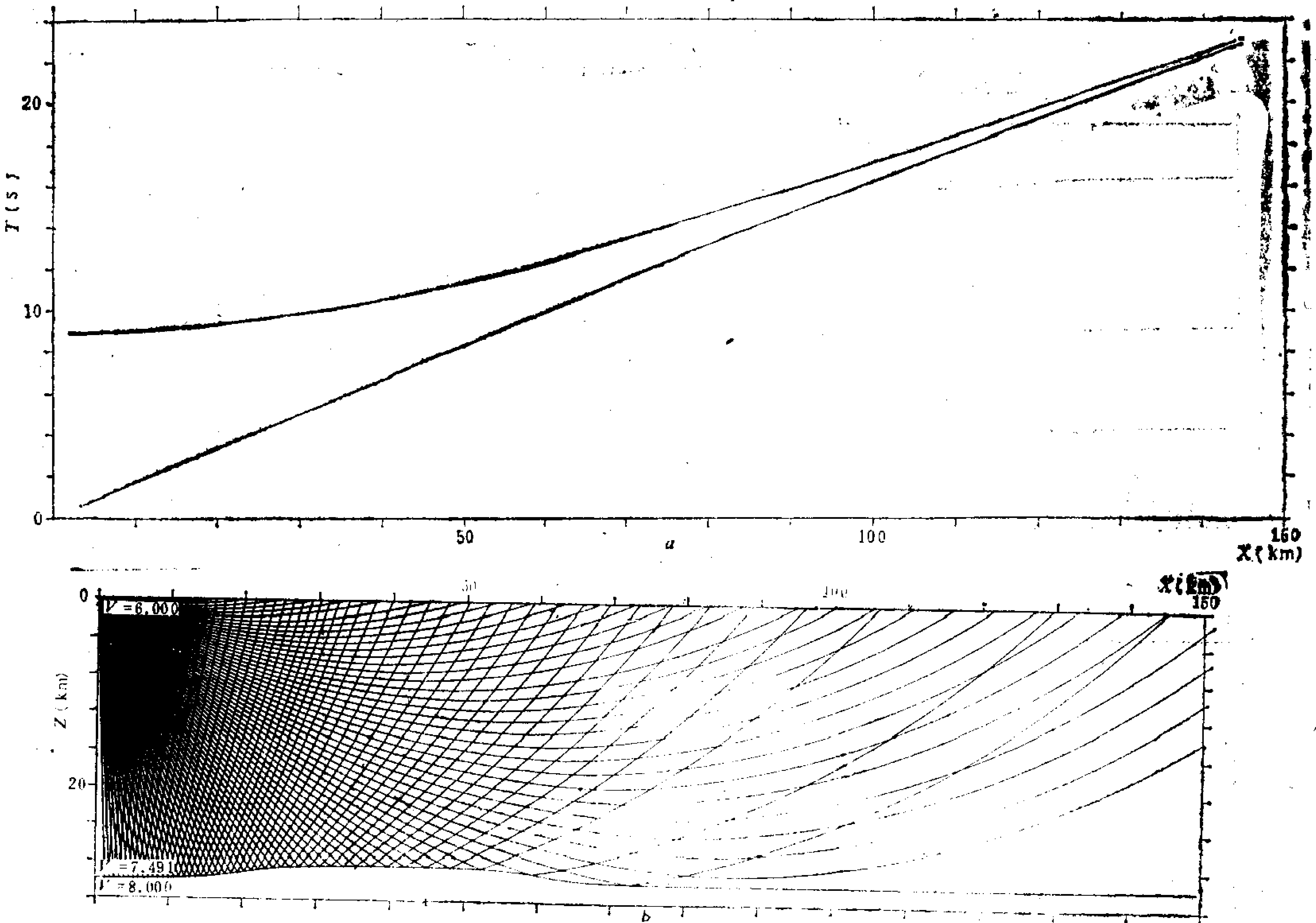



图 10-2-13 非均匀介质中反射波和回折波时距曲线和射线分布图

a.时距曲线图； b.射线分布图

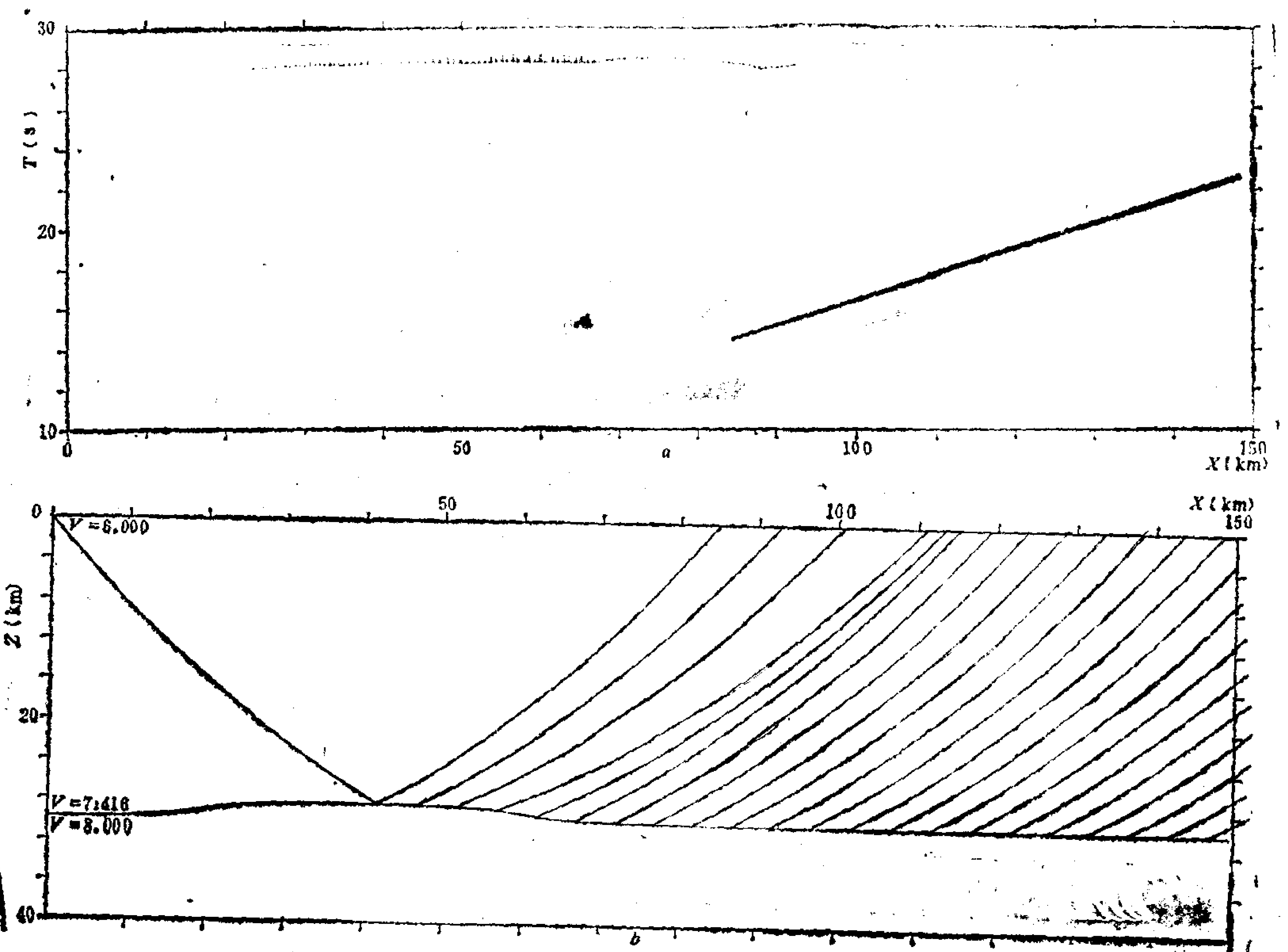



图 10-2-14 非均匀介质中折射波时距曲线和射线分布图

a,时距曲线图； b,射线分布图

虑了区间约束和特定约束条件，根据实际情况，对不同的反演参数分别给不同的初始阻尼系数。在多层介质中，下层的走时要受上面各层界面和速度分部的影响，因此一般是由上而下分层反演。有关阻尼最小二乘法的原理已在前面讨论过了，现仅就射线追踪反演中有关问题讨论如下。

根据实测走时曲线的特点合理地选择反演参数很重要，它不但影响到收敛的速度，而且影响到是否能收敛的问题。例如把一个本来是水平常速层的情况，误作为一个上凸或下凹的界面来进行反演，那么反演参数由两个一下就增加到八个，这样会大大增加每次迭代的时间；反之，如果把本来是一个上凸界面误作为水平层来进行反演，那么会发生不收敛而使计算失败。

除了应当合理地选择反演参数外，还应十分重视反演参数初值的选择，它不但影响到收敛的速度，而且关系到是否会形成假收敛的问题。一般说来，在极小点附近 $\phi$ 的等值面比较接近于椭球面，当初值取得靠近极小点时，收敛较快。当初值取得远离极小点时，$\phi$ 函数等值面一般变得比较复杂，特别是当等值面拉长和弯曲时，收敛速度变慢。阻尼最小二乘法一般只适于 $\phi$ 函数是单极值的情况，而实际问题中的 $\phi$ 函数等值面比较复杂，可能出现多个极小点，因此初值给得不当，不一定能收敛到真正的极小点，造成假收敛，得到错误的结果。

反演的多解性降低了反演结果的可靠性和实用价值。数学技巧不能弥补地球物理测量的信息不足，只有设法补充信息，方法之一就是在反演时加入约束条件。约束的方法很多，本程序采用了区间约束、特定约束，界面不能交叉等约束条件。区间约束就把反演参数的变化范围限制在某个区间之内，即求出的改正向量加到对应反演参数上，如果某个参数超过约束区间则不加对应的改正向量，使参数始终在约束区间内变化。特定约束是指把某些通过其它手段得到的比较可靠的参数（例如通过钻孔得到的界面深度）不参加反演。

射线追踪反演参数的权重不一样，每次迭代的修改量要求不同，对不同的参数给以不同的初始阻尼较为合适。这样会加快收敛速度，并且使反演结果比较合理。

反演参数之间的量纲和非线性程度不一样，Jacobi 矩阵偏导数的增量值也不能取同一个值。例如将偏导数增量统一取为 $10^{-5}$，在有些情况下对界面参数 $X_{0ki}$ 是合适的，但对界面参数 $b_{ki}$ 就不合适，常常出现机器零而使运算溢出。

上面分别对反演参数及其初值、约束条件、阻尼系数进行了粗略的讨论，它们之间并不是孤立的，而是相互制约、彼此影响的。某个参数不参加反演，等效于阻尼系数取无穷大；如果取的初值离真值不太远，那么约束区间应取得比较小。总之应当按照实测时距曲线的特点以及已经掌握的地下构造的情况，综合分析，统筹考虑反演参数、初值、约束条件、阻尼系数等。

图 10-3-1 是利用约束阻尼最小二乘法编写的射线追踪反演程序将华北某地区人工地震剖面实测反射波走时资料作常速弯曲界面反演的例子。设坐标系的原点位于炮点。上图是实测走时与理论走时的拟合情况图，带点的实线是实测走时曲线，不带点的实线是理论走时曲线。下图是反演后的速度和界面情况图。第一个界面是康氏界面，记为 $C_1$，在离开炮点约 10—20 公里的地方缓慢下凹，第二个界面是玄武岩层内的一个反射界面，在离炮点约10公里的地方缓慢下倾，第三个界面是莫霍界面，在离开约40公里的地方逐渐下倾。

## § 10-4　射线追踪反演初值的确定

上一节讨论了初值的好坏对射线追踪反演结果影响很大。一般说来，用椭圆法或时间场的反演结果作为射线追踪反演的初值效果较好。本节将介绍椭圆法及时间场法反演，以及如何利用它们的反演结果作为射线追踪反演的初值进一步反演，以获得符

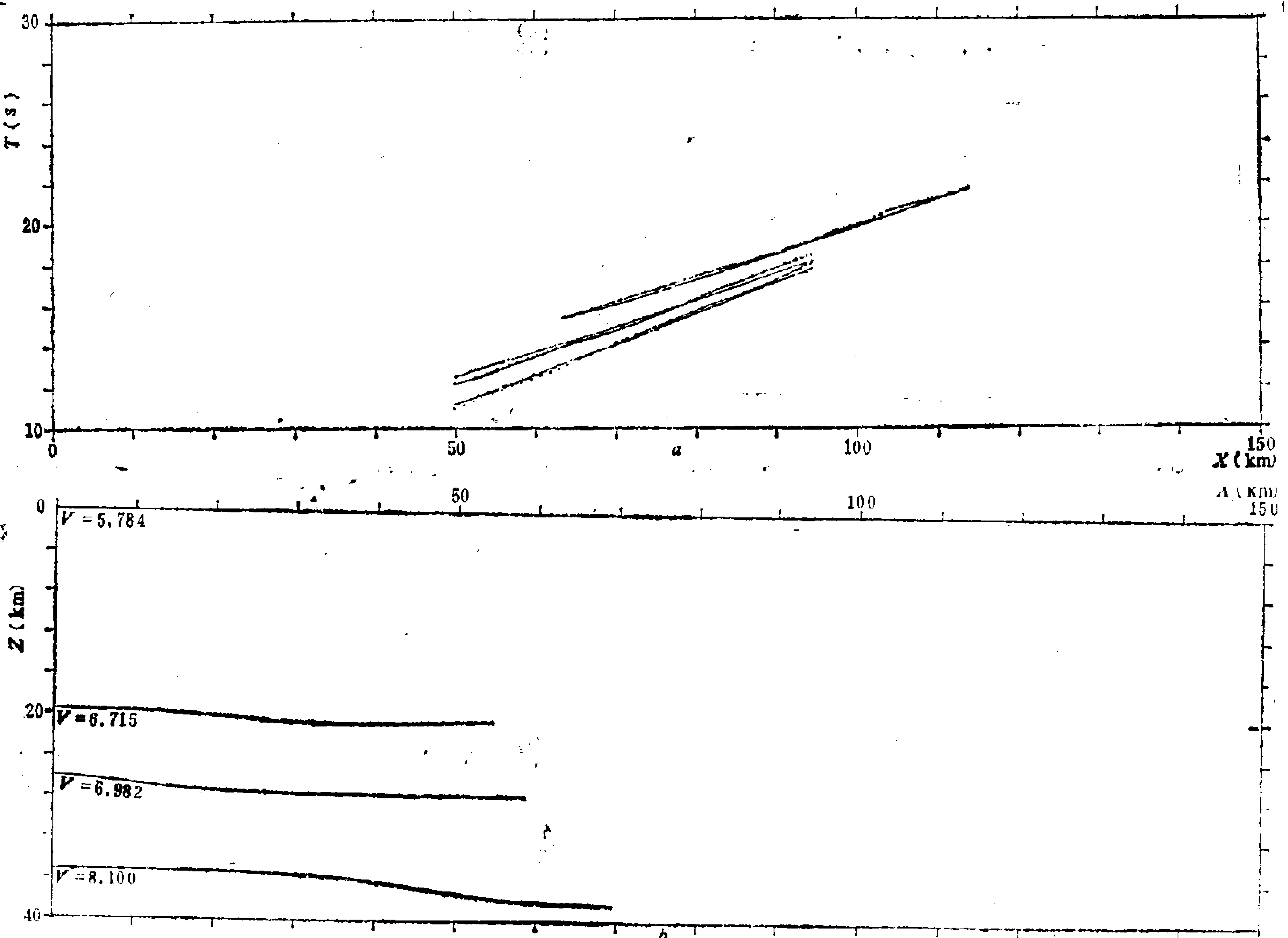



图10-3-1 人工地震剖面××炮点反演结果

a 时距曲线图　　b.反演结果

合客观实际的反演结果。

## （一）椭圆法反演

椭圆法反演是根据在均匀介质中以炮点和测点分别作为椭圆的两个焦点，则反射点必在其椭圆上的原理（参看图10-4-1），求出所有测点对应的椭圆，然后计算出这些椭圆簇的包络线，该包络线即为所求的界面。

第 $i$ 个测点的椭圆方程为

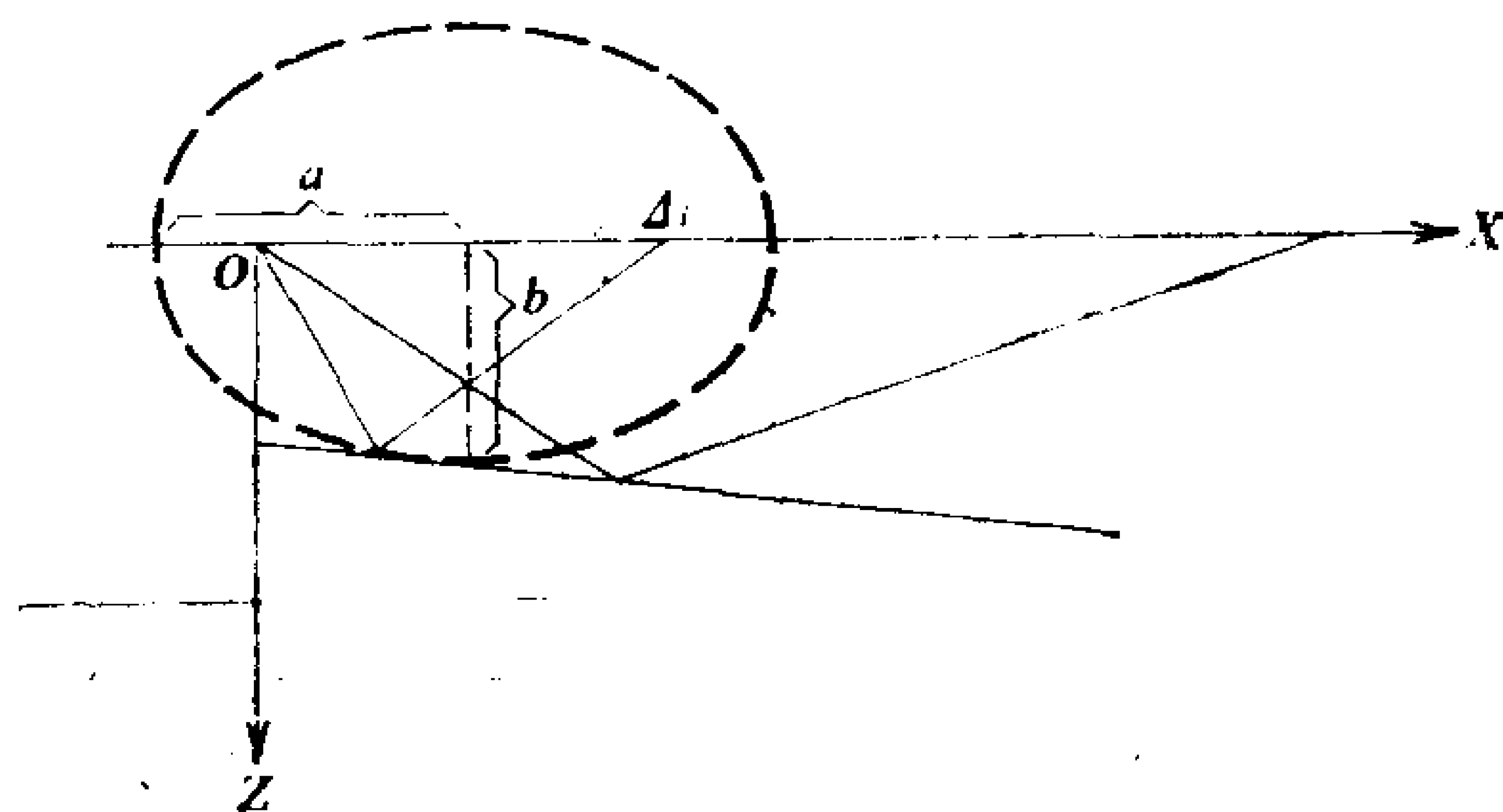



图10-4-1 椭圆反演示意图

$$\frac{(X-L_i)^2}{a_i^2}-\frac{Z^2}{b_i^2}=1$$

式中 $L_i=\frac{\Delta_i}{2}$，$\Delta_i$ 是第 $i$ 个测点的炮检距。$a_i=\frac{1}{2}Vt_i$，$t_i$ 是第 $i$ 个测点的到时；$V$ 是有效速度，用 $\Delta_i^2-T^2$ 法从时距曲线上求取，即 $V=\frac{1}{N-1}\sum_{i=1}^{N-1}\sqrt{(\Delta_{i+1}^2-\Delta_i^2)/(t_{i+1}-t_i)^2}$；$N$ 为测点个数；$b_i=\sqrt{a_i^2-L_i^2}$。

椭圆簇的包络线可以这样来求取：求出相邻两个椭圆的公切线，然后各个切点用样条函数连接起来。

用计算机求公切线有好几种方法，下面仅介绍一种常用的方

法——逐次逼近法。

逐次逼近法首先确定相邻两椭圆极值点连线的斜率 $K$（请参看图10-4-1）

$$K=(Z_{i+1}-Z_i)/(L_{i+1}-L_i)$$

当$K\geqslant0$时，令 $SIGN=-1$；当$K<0$时，令 $SIGN=1$

计算第 $i$ 个椭圆和第 $i+1$ 个椭圆的切线斜率 $K_i$ 和 $K_{i+1}$；

$$K_i=\frac{b_i(X_i-L_i)}{a_i\sqrt{a_i^2-(X_i-L_i)^2}}$$

$$K_{i+1}=\frac{b_{i+1}(X_{i+1}-L_{i+1})}{a_{i+1}\sqrt{a_{i+1}^2-(X_{i+1}-L_{i+1})^2}}$$

式中：$X_i=L_i+\Delta X\cdot m\cdot SIGN\quad(m=0,1,2,\cdots,n)$

$$X_{i+1}=L_{i+1}+\Delta X\cdot K\cdot SIGN\quad(k=0,1,2,\cdots,n)$$

$$Z_i=\sqrt{[1-(X_i-L_i)^2/a_i^2]\cdot(a_i^2-L_i^2)}$$

$$Z_{i+1}=\sqrt{[1-(X_{i+1}-L_{i+1})^2/a_{i+1}^2]\cdot(a_{i+1}^2-L_{i+1}^2)}$$

$$\Delta X=L_i/n$$

$m$ 每增加一个步距，$K$从0到 $n$ 循环一次。

若 $|K_{i+1}-K_i|<\varepsilon$（$\varepsilon$ 为所取的精度，一般取0.01）则认为第 $i$ 个椭圆与第 $i+1$ 个椭圆的切线重合，是这两个椭圆的公切线，切点坐标分别为（$X_i$，$Z_i$）和（$X_{i+1}$，$Z_{i+1}$）。用样条曲线(二次或三次)把这些切点连接起来，即得到所求作的反射界面。

前面已讲过，对于反演用反正切函数表示的多项式来描述界面较好。下面简述从求作的样条曲线表示的界面如何用计算机确定。用反正切函数多项式表示该界面的参数计算步骤(参看图10-4-2)。

（1）计算最初两点的斜率$K_{1,2}$，则 $a_{0k}=K_{1,2}$。

（2）划分反正切曲线段。当$|K_{jm}-K_0|>\varepsilon$就分成一个反正切曲线段。$K_{jm}$表示第 $j$ 个反正切曲线段 $m$个测点中的最大斜率，$\varepsilon$表

示分段精度。

(3) 计算第 $j$ 段反正切曲段的落差 $\Delta Z_i$ 和中心点 $X_{0j}$。设第 $j$ 段反正切曲斜率最大的两点为 $A$ 和 $B$，坐标分别为 $(X_{jA}, Z_{jA})$，$(X_{jB}, Z_{jB})$，显然

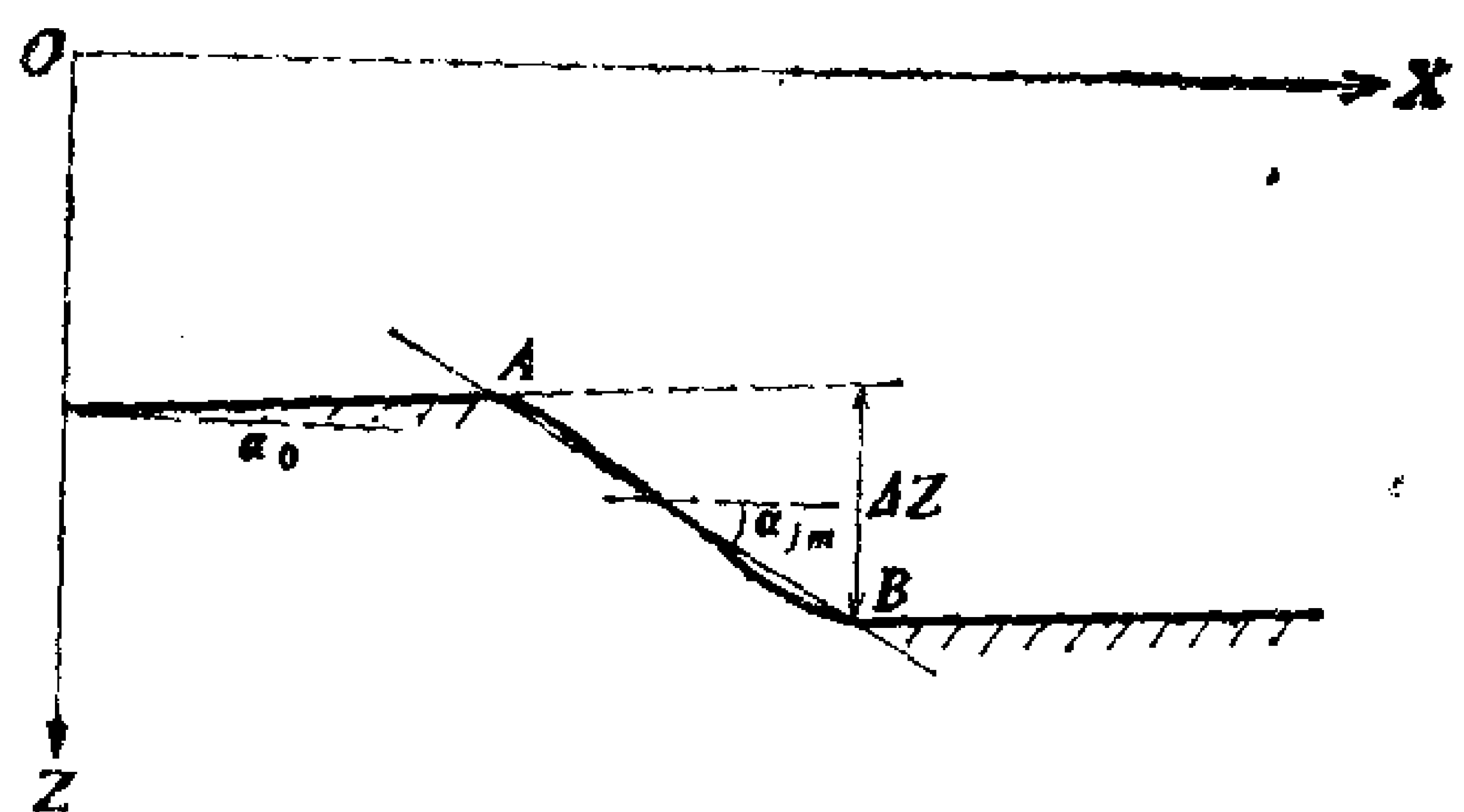


图 10-4-2 确定反正切参数示意图

$$\Delta Z_j = Z_{jB} - Z_{jA}$$
$$X_{0j} = X_{jA} + (X_{jB} - X_{jA})/2$$

(4) 计算反正切曲线的陡度 $b_j$。用样条插值求出该段 $\frac{\Delta Z_i}{4}$ 对应的 $X_j$，则 $b_j = |X_i - X_{0j}|$。

## （二）时间场法反演

根据反射界面上入射波等于时线与反射波等时线相等的原理，分别求出入射波等时线和反射波等时线，然后计算它们的交点，再把这些交点用样条函数连结起来，便得到所要求的地下界面。

时间场法的具体计算步骤如下：

(1) 计算反射波等时线。在均匀介质中，计算某时刻 $t_i$ 等时线的方法是：分别以测点（或者时差为 $\Delta t$ 的各个点）$X_{i+1}$，$X_{i+2}$，…，$X_{n-i}$ 为圆心，

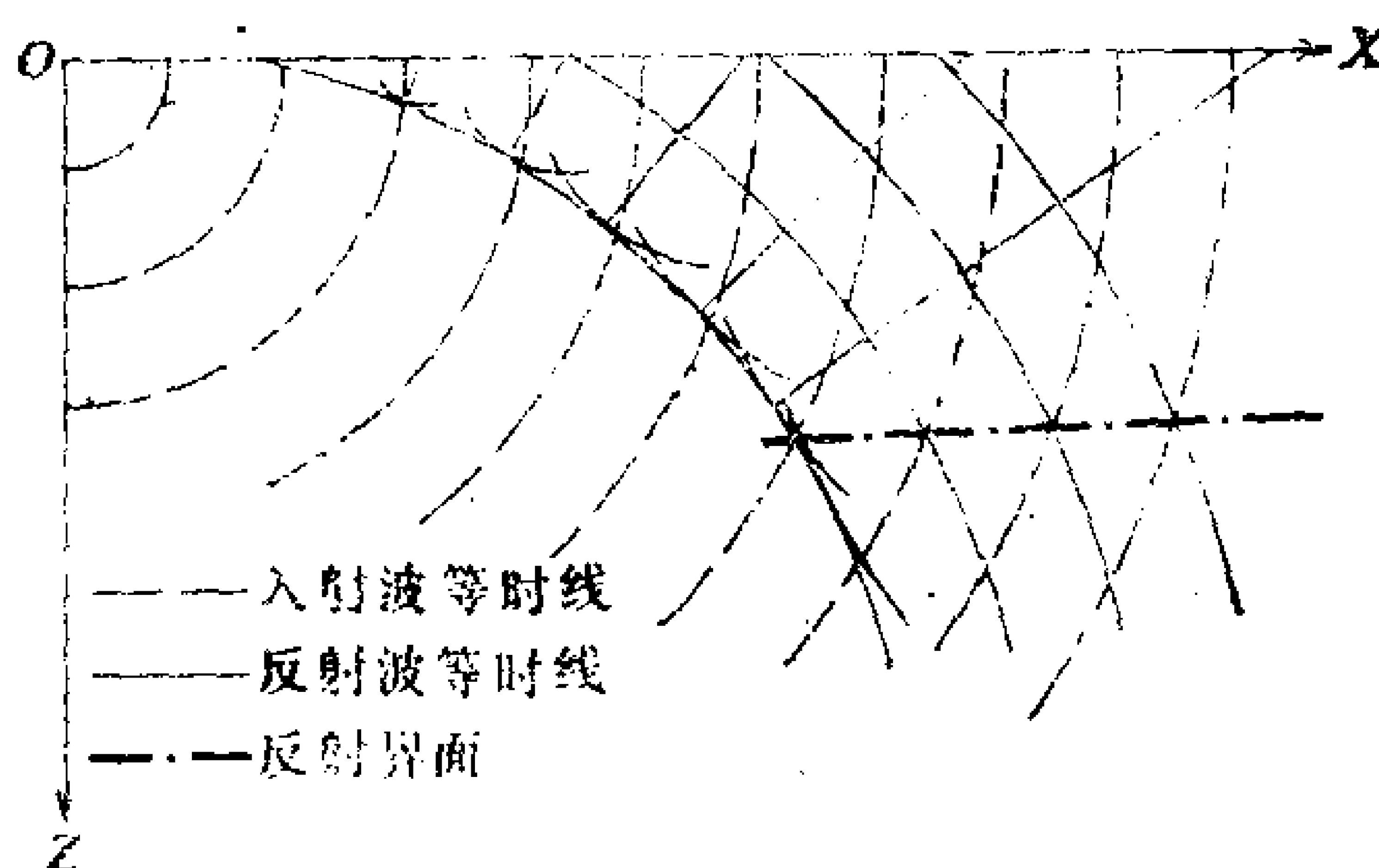


图 10-4-3 时间场法反演示意图

以 $V(t_{i+1}-t_i)$，$V(t_{i+2}-t_i)$，…，$V(t_{n-i}-t_i)$ 为半径作圆，这些圆的包络线就是 $t_i$ 时刻的波前。包络线可以这样来

作：计算相邻两圆的公切线，各个切点用样条曲线连接起来，近似地认为是诸圆的包络线。相邻两圆公切线切点的坐标可按下式计算（参看图10-4-4）：

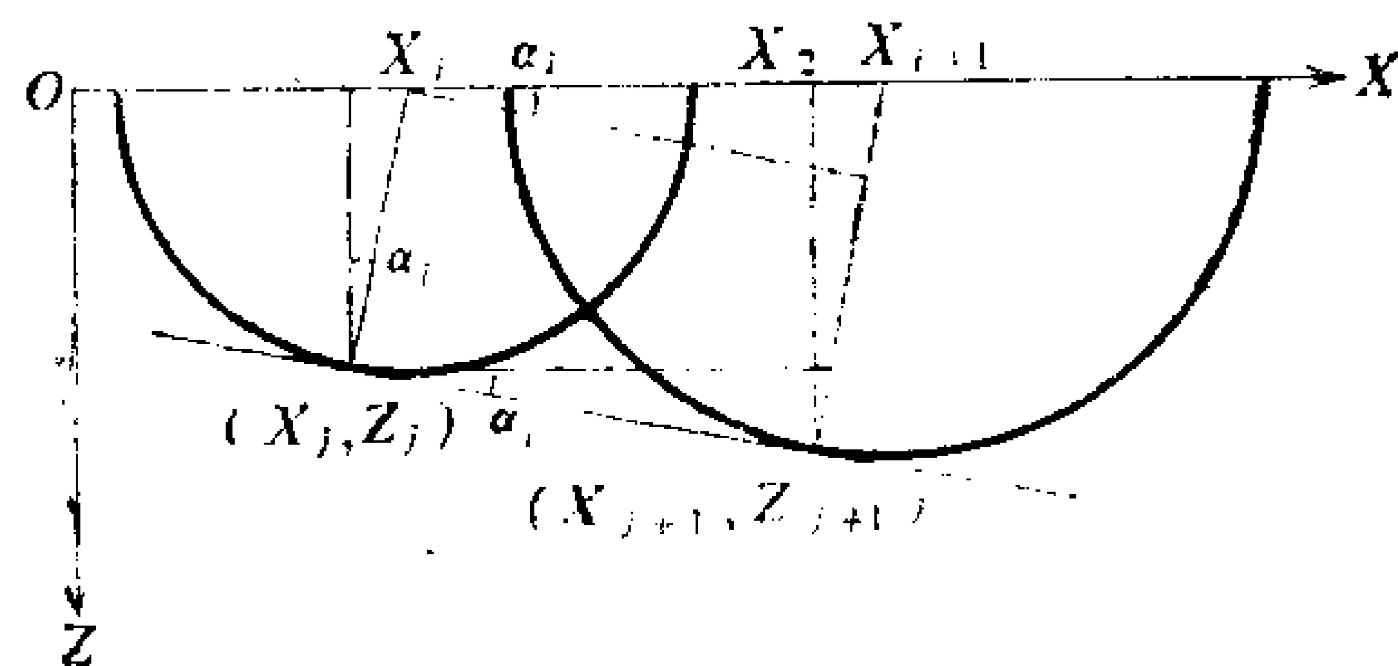



图 10-4-4 计算相邻两圆的公切线

$$X_j = X_i - R_i \sin\alpha_i$$

$$Z_j = R_i \cdot \cos\alpha_i$$

$$X_{j+1} = X_{i+1} - R_{i+1} \sin\alpha_i$$

$$Z_{j+1} = R_{i+1} \cos\alpha_i$$

式中 $\alpha_i = \arcsin[(R_{i+1} - R_i)/(X_{i+1} - X_i)]$

（2）计算$t_i$时刻入射波的等时线。在均匀介质中，它是以炮点为圆心，$R = Vt_i$为半径的圆。

（3）计算$t_i$时刻入射波等时线与反射波等时线的交点，把这些交点用样条曲线连结起来，即为所求的反射界面。

用时间场法反演得到的反射界面作为射线追踪反演的初值，计算反正切函数多项式的各个参数与椭圆法相同。

# §10-5 射线追踪正、反演计算程序

## （一）程序功能

正演部分：用射线追踪计算均匀或非均匀介质中任意界面的反射波、首波和回折波走时，可以用绘图机自动绘出射线分布图和时距曲线。

反演部分：用约束阻尼最小二乘法根据反射波或首波走时反演地壳和上地幔的速度分布和界面起伏情况。在走时实际上是单值，模型初值、约束条件、初始阻尼系数都给得比较合理的条件

下能获得较好的结果。即可以由上而下依次先进行折射波后进行反射波反演，也可以对某一层或某几层的某几个参数进行折射波或反射波反演。用绘图机绘出反演结果，即地下界面起伏情况和速度分布，理论走时与实测走时的拟合情况。

## （二）主要标识符

MTAN——界面形如反正切曲线的个数。

MG——反演首波的层数（只作正演时为模型的层数）。

MF——反演反射波的层数（只作正演时为零）。

M——测点总数（只作正演时为零）。

MZ——接收点总数（只作正演时为零）。

IO
- =0 炮点在绝对坐标系的原点，且接收点在坐标系的正方向。
- =1 炮点不在绝对坐标系的原点，或接收点不在坐标系的正方向。

ME
- <0 只反演首波。
- =0 只反演反射波，
- >0 先反演首波，后反演反射波。

KC
- =0 输出是折合走时。
- =1 输出是一般走时。

KX 折合速度。

KY
- =0 不输入约束条件数组（此时利用程序中的束条件）。
- =1 输入约束条件。

KD
- =0 不输入阻尼系数数组（此时利用程序中给的阻尼系数）。
- =1 输入阻尼系数数组。

KJ
- =0 不输入 Jacobi 矩阵偏导数增量数组（此时利用程序中给的数值）。
- =1 输入 Jacobi 矩阵偏导数增量数组。

KI $\begin{cases} =0 & \text{不打印中间计算结果。} \\ =1 & \text{打印详细的中间计算结果。} \\ =5 & \text{打印反演计算中间结果。} \end{cases}$

KFR $\begin{cases} =0 & \text{不画射线图。} \\ =1 & \text{画射线图(此时 KZ 必须填 1)。} \end{cases}$

KFT $\begin{cases} =0 & \text{不画时距曲线。} \\ =1 & \text{画时距曲线(此时 KZ 必须填 1)。} \end{cases}$

KFZ $\begin{cases} =0 & \text{不画反演结果图。} \\ =1 & \text{画反演结果图(此时 KZ必须填 0)。} \end{cases}$

KSP——最大迭代次数。

TAPE——希望射线到达地面的间距。

KZ $\begin{cases} =0 & \text{作反演。} \\ =1 & \text{只作正演。} \end{cases}$

NNS——射线的条数(最好取 NNS=(XMAXR-XMIXR)/(2*TAFE)+1).

GLX——射线图横轴的长度(cm)。
LX——横轴的刻度间隔数。
GLY——射线图纵轴的长度(cm)。
LY——纵轴的刻度间隔数
XMINR——坐标原点的横坐标值。
XMAXR——横坐标的最大值。
YMINR——坐标原点的纵坐标值。
YMAXR——纵坐标的最大值。

（以上）反演时为地下界面图对应的图框简变。

GLXT——时距图横轴的长度(cm)。
LXT——横轴刻度的间隔数。
GLYT——时距图纵轴的长度(cm)。
LYT——纵轴刻度的间隔数。
XMINT——坐标原点的横坐标值。
XMAXT——横坐标的最大值。

（以上）反演时为时距曲线拟合情况图对应的图框简变。

YMINT——坐标原点的纵坐标值。

YMAXT——纵坐标最大值。

SH1——炮点的横坐标。

SH2 $\begin{cases} =1 & \text{测点在正方向。} \\ =-1 & \text{测点在反方向。} \end{cases}$

AC(MM1，MAC)——模型参数数组。

MM1——界面个数。

MAC——参数个数，MAC=MTAN*3+7。

AC(K+1,1)——第 $K$ 个界面的下速度。

AC(K+1,2)——第 $K+1$ 个界面的上速度。

AC(K+1,3)——第 $K+1$ 个界面的下速度。

AC(K+1,4)——第 $K$ 层的横向速度梯度。

AC(K+1,5)——第 $K$ 层的纵向速度梯度。

AC(K+1,6)——绝对坐标系原点到第 $K+1$ 个界面的距离。

AC(K+1,7)——第 $K+1$ 个界面的斜率。

AC(K+1，3*I+5)——第 $K+1$ 个界面的第 $I$ 个反正切曲线的顶、底距离。

AC(K+1,3*I+6)——第 $K+1$ 个界面的第 $I$ 个反正切曲线中点横坐标。

AC(K+1，3*I+7)第 $K+1$ 个界面的第 $I$ 个反正切曲线的四分之一高作顶线的垂线，该反正切曲线中点到此垂线的距离。

如果炮点不在坐标原点，模型参数数组为AC1(MM1,MAC)，各参数的意义与 AC(MM1，MAC) 相同。

IH(MD) 各层接收点号数组。MD 为数组元素个数，MD=MG+MF+MZ。该数组的数据填写方法是由上而下先折射波后反射波依次填写各层走时接收点总数，再由上而下先首波后反射波填写各层走时接收点号。

PGJ(M) 炮检距数组。$M$ 表示测点个数，该数组按炮检距的大小，由小到大填写。

IX(MM1, MAX) 反演参数信息数组。MM1 表示界面个数，MAX 等于反演参数加一。该数组的第一列分别为各层反演参数的总个数。其它各列元素分别与参数数组元素对应，某个参数参加反演填 1 ，不参加反演填 0 。

例如某地壳模型有三层，地面水平，第一层界面是由两个反正切曲线构成的隆起界面，速度随深度线性变化，第二和第三层的界面都是形如一个反正切曲线的界面，均为常速层。假设让所有的参数都参加反演，显然第一层反演参数有 9 个，第二层和第三层有 5 个。如果只作反射波反演，则反演参数数组应为：

0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
9, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1,
5, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0,
5, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0

AY1(MM1, MAC) 约束条件数组下界。

AY2(MM1, MAC) 约束条件数组上界。

DDQ(MAC) 阻尼系数初值数组。

XDJ(MAC) Jacobi 矩阵偏导数自变量增量数组。上面各数组的数组元素大小 MM1、MAC 与参数数组 *AC* 中的数组元素大小MM1、MAC相同。

## （三）各子程序功能及虚元说明

1. FIGR：绘射线图。

2. FIGZ：绘界面图。

3. FIGT(TX, NNF, K)：绘反射波、折射波时距曲线。

TX——二维数组，存放时距数值。

NNF——实际计算反射波或首波的射线条数。

K——第 *K* 层。

4. FIGTR(TX, NNF, K)：绘回折波时距曲线。

TX——二维数组，存放回折波时距数值。

NNF——实际计算回折波的射线条数。

5. FIGT1(PX, PT, NNF, K)：绘实测走时与理论走时的拟合情况图。

PX——一维数组，存放炮检距数值。

PT——一维数组，存放走时

NNF——PX、PT数组元素的个数。

6. RAH(GLX, IX, GLY, IY, GH, KOM, XMIN, XMAX, YMIN, YMAX)：绘各种图形的坐标系。

GLX——横坐标轴的长度（cm）。

IX——横坐标轴的刻度间隔数。

GLY——纵坐标轴的长度（cm）。

IY——纵坐标轴的刻度间隔数。

GH——数字的高度。

KQM——纵坐标轴的方向，向上为0，向下为1。

XMIN——坐标原点的横坐标数值。

XMAX——横坐标端点的数值。

YMIN——坐标原点的纵坐标数值。

YMAX——纵坐标端点的数值。

7. MOVE(N1, TX, TX1)：将炮点坐标系的时距值变换成绝对坐标系的时距值。

N1——计算的射线条数。

TX——二维数组 存放炮点坐标系的时距值。

TX1——二维数组，存放绝对坐标系的时距值。

8. Z(X, K)：计算界面的纵坐标。

X——横坐标值。

K——第 $K$ 层。

9. XY(C)：计算每跨一角度步长C，射线端点的坐标和走时。

C——角度步长。

10．SEARC(S1，S2)：二分法求界面交点。

S1——初值。

S2——射线前进的步长。

11．ANGLE(K)：计算射线与界面交点处的界面倾角、反射角、临界角、折射角、射线与水平线的夹角等。

K——第 $K$ 层。

12．TRACE(K)　在第$K$层内具体地进行射线追踪，计算射线的坐标和走时。

K——第 $K$ 层。

13．UPTXQ(N1，NN1，TX，TX1)　计算目的层的上行射线。

N1——计算第 N1 条射线。

NN1　存放于数组 TX、TX1 的第 NN1 条射线。

TX——二维数组，存放炮点坐标系的时距值。

TX1——二维数组，存放绝对坐标系的时距值。

14．UPTXZ(N1，NN1，TX，TX1）计算目的层以上各层的时距值，各虚元的意义同 UPTXTO。

15．ZAH(N，X，T，U，TF)　一元三点插值。

N——插值结点的个数。

X——一维数组，存放 N 个插值结点。

T——一维数组，存放 N 个插值结点的函数值。

U——插值点。

TF——插值结果。

16．AQ(N1，F1，PX，IM，TXX，TXT)：计算测点的理论走时。

N1——插值结点的个数。

FT——一维数组，存放测点理论走时。

PX——一维数组，存放测点横坐标。

IM——测点个数。

TXX——一维数组，存放炮检距。

TXT——一维数组，存放走时。

17. ZYFT(PT, PX, IM) 控制整个追踪过程，计算理论走时。

PT——一维数组，存放测点的理论走时。

PX——一维数组，存放测点的横坐标。

IM——测点个数。

18. BCAJ (CAJ, FT, FTQ, PX, NNX, IM) 求Jacobi矩阵。

CAJ——二维数组，存放Jacobi矩阵的数值。

FT——一维数组，存放反演参数加上一个增量后计算的理论走时。

FTQ——一维数组，存放反演参数不加增量计算的理论走时。

PX——一维数组，存放测点的横坐标。

NNX——反演参数加1，Jacobi矩阵的维数。

IM——测点个数。

19. OF(FT, F, DPT, PT, IM) 计算残差，残差平方和以及平均绝对误差。

FT——一维数组，存放理论走时。

F——残差平方和。

DPT——一维数组，存放残差。

PT——一维数组，存放实测走时。

IM——测点个数。

20. OUTZ 防止界面交叉。

21. MAQT(FT, FTO, PX, PT, DPT, IM, DDP, R1, XO, X, AW, NX, AZE, NNX, CAJ) 用理论走时拟合实测走时。

FT——一维数组，存放反演参数加上改正向量后计算的走

时。

FTO——一维数组，存放反演参数向量不加改正向量计算的走时。

PX——一维数组，存放测点横坐标。

PT——一维数组，存放实测走时。

DPT——一维数组，存放残差。

IM——测点个数。

DDP——一维数组存放的参数的初始阻尼系数。

R1——一维数组，存放 AZE 矩阵的对角元素的平方根。

XO——一维数组，存放反演参数的数值。

X——一维数组，存放反演参数加上改正量后的数值。

AW——一维数组，存放改正向量。

NX——反演参数个数。

AZE——增广矩阵。

NNX——反演参数加一。

CAJ——Jacobi 矩阵。

### (四) 程序结构、主程序及主要子程序框图

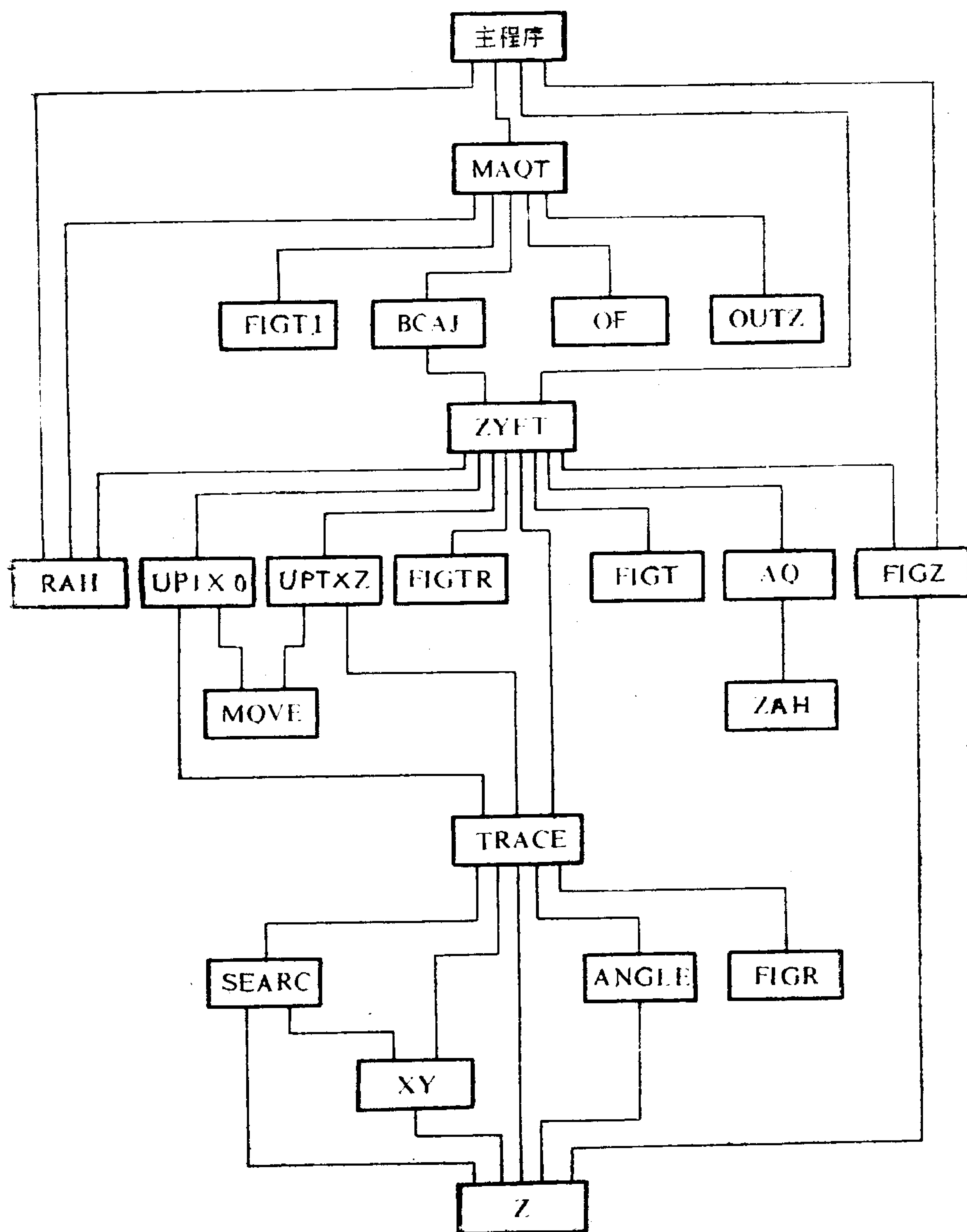



图10-5-1 主程序与子程序之间的检用关系

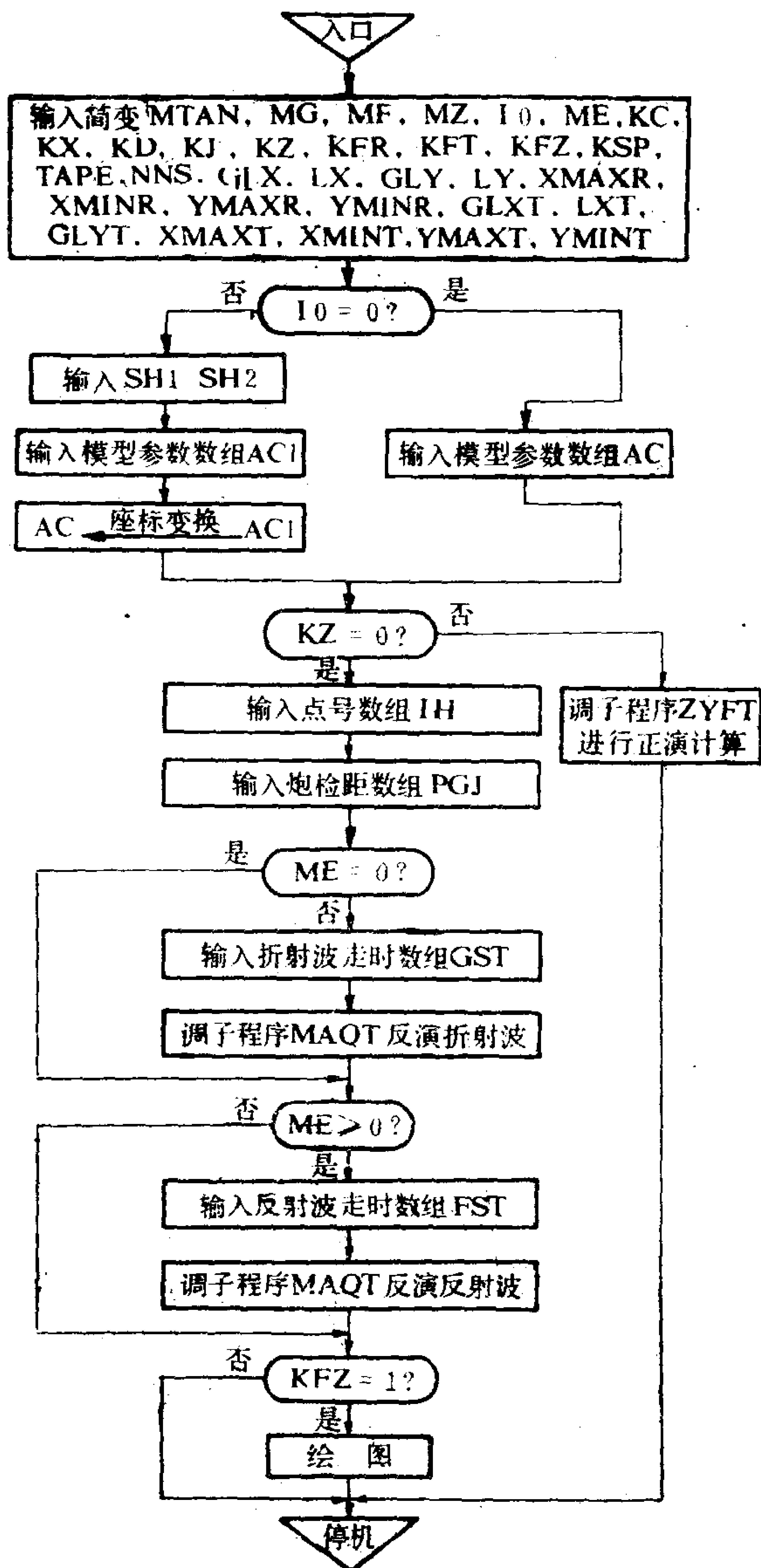

入口
输入简变MTAN，MG，MF，MZ，I0，ME，KC，KX，KD，KJ，KZ，KFR，KFT，KFZ，KSP，TAPE，NNS，GLX，LX，GLY，LY，XMAXR，XMINR，YMAXR，YMINR，GLXT，LXT，GLYT，XMAXT，XMINT，YMAXT，YMINT
否
I0 = 0?
是
输入SH1 SH2
输入模型参数数组AC1
输入模型参数数组AC
AC ← 座标变换 AC1
否
KZ = 0?
是
输入点号数组IH
调子程序ZYFT进行正演计算
输入炮检距数组PGJ
是
ME = 0?
否
输入折射波走时数组GST
调子程序MAQT反演折射波
否
ME > 0?
是
输入反射波走时数组FST
调子程序MAQT反演反射波
否
KFZ = 1?
是
绘 图
停机

图10-5-2 主程序框图

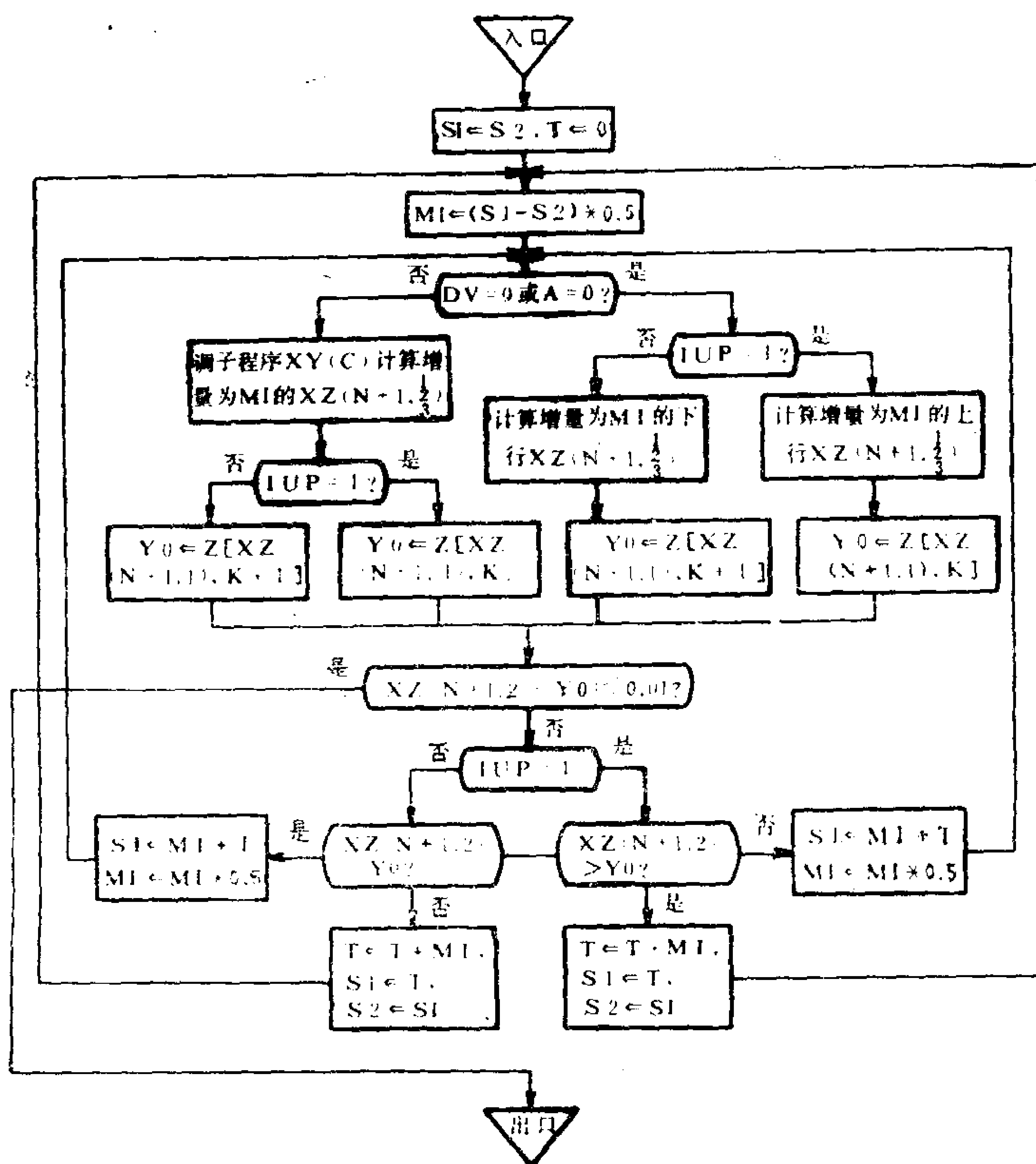
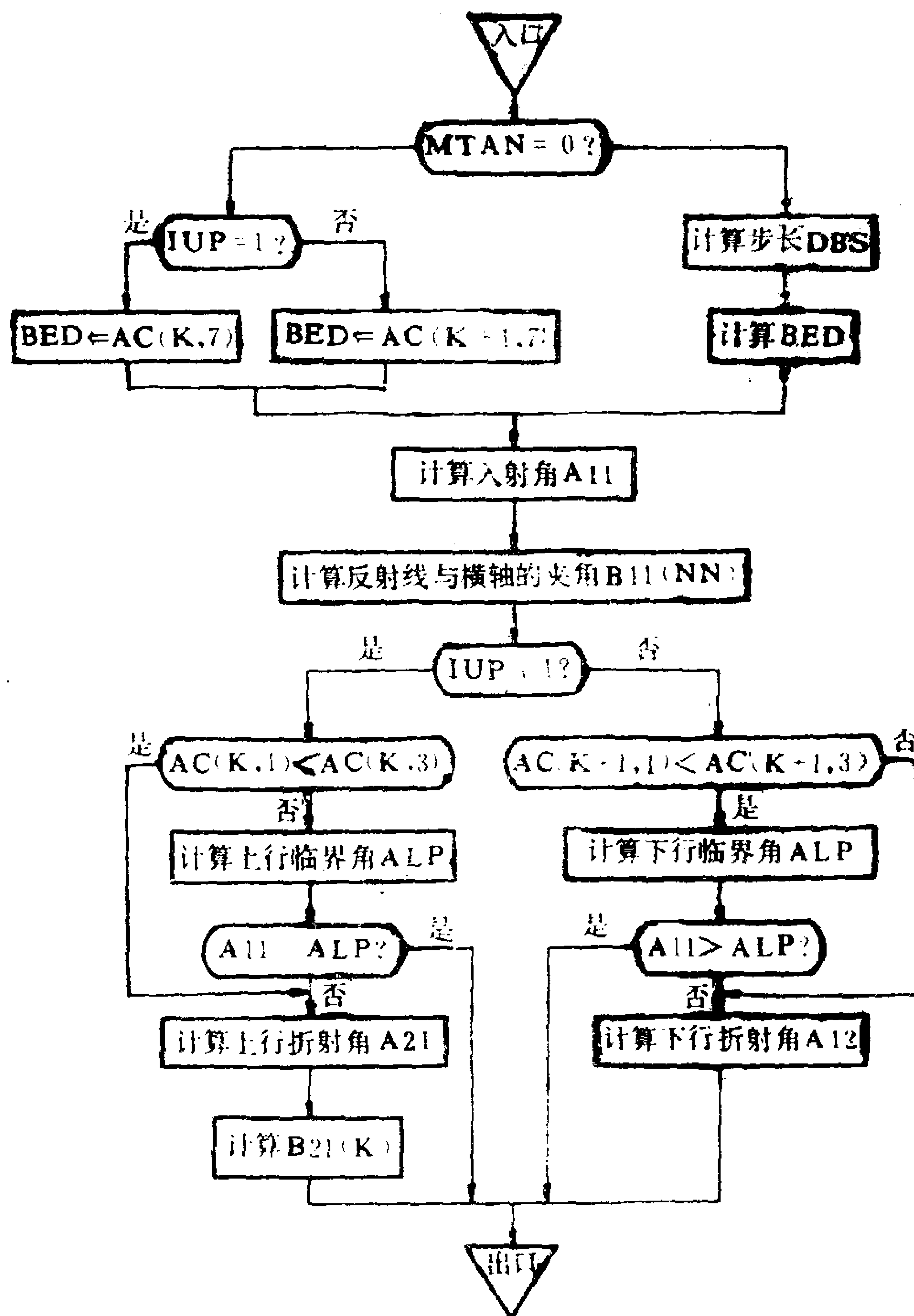



图10-5-3 SEARC子程序框图

注：XZ(N+1,1)：第N+1次追踪的横坐标，XZ(N+1,2)：第N+1次追踪的纵坐标，XZ(N+1，3)：第N+1次追踪的走时，下同




注：BED：射线与界面交点处的倾角

B21(K)：第K层上行折射线与横轴的夹角

图10-5-4 ANGLE子程序框图

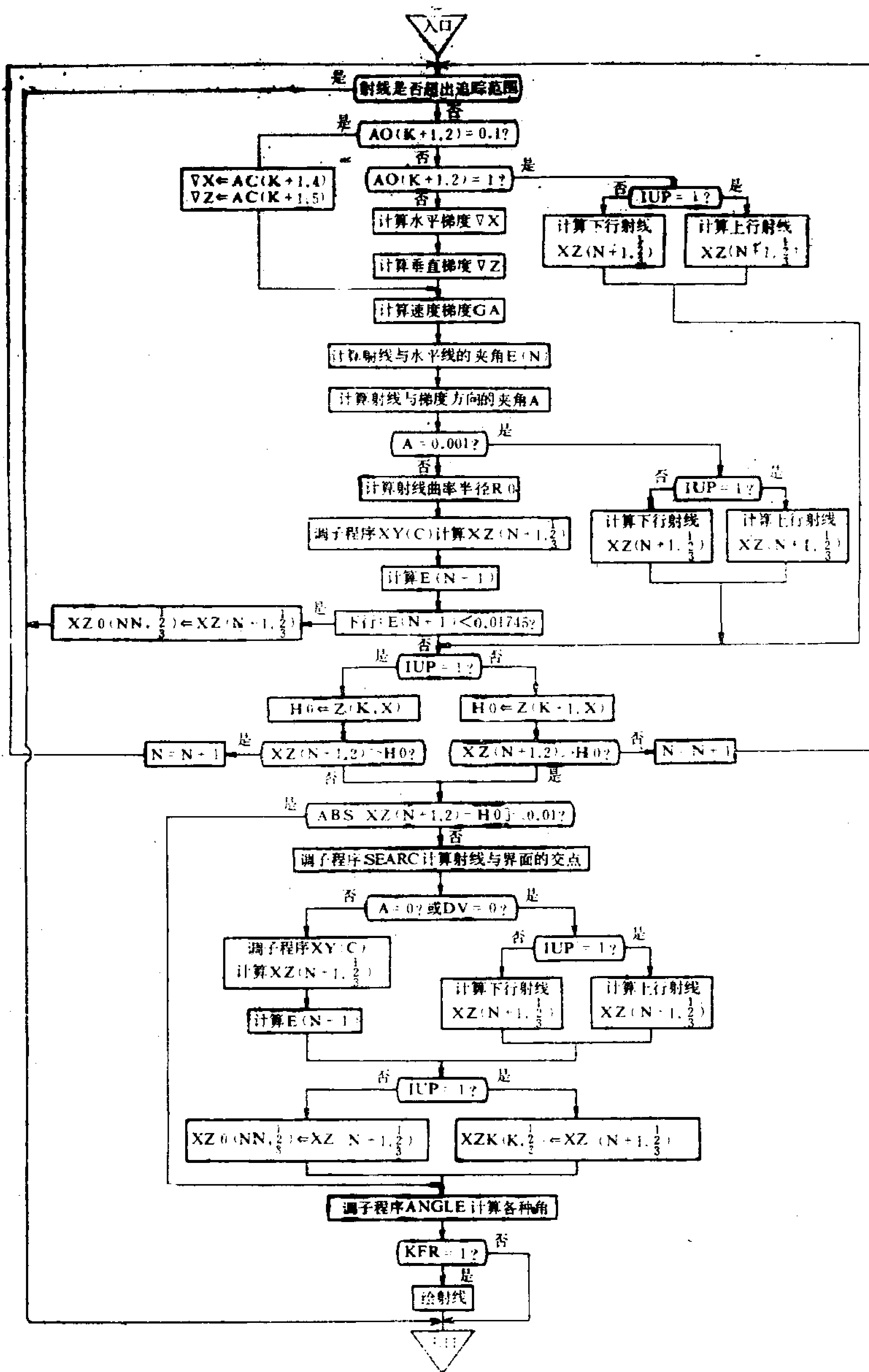



图10-5-5　TRACE子程序框图

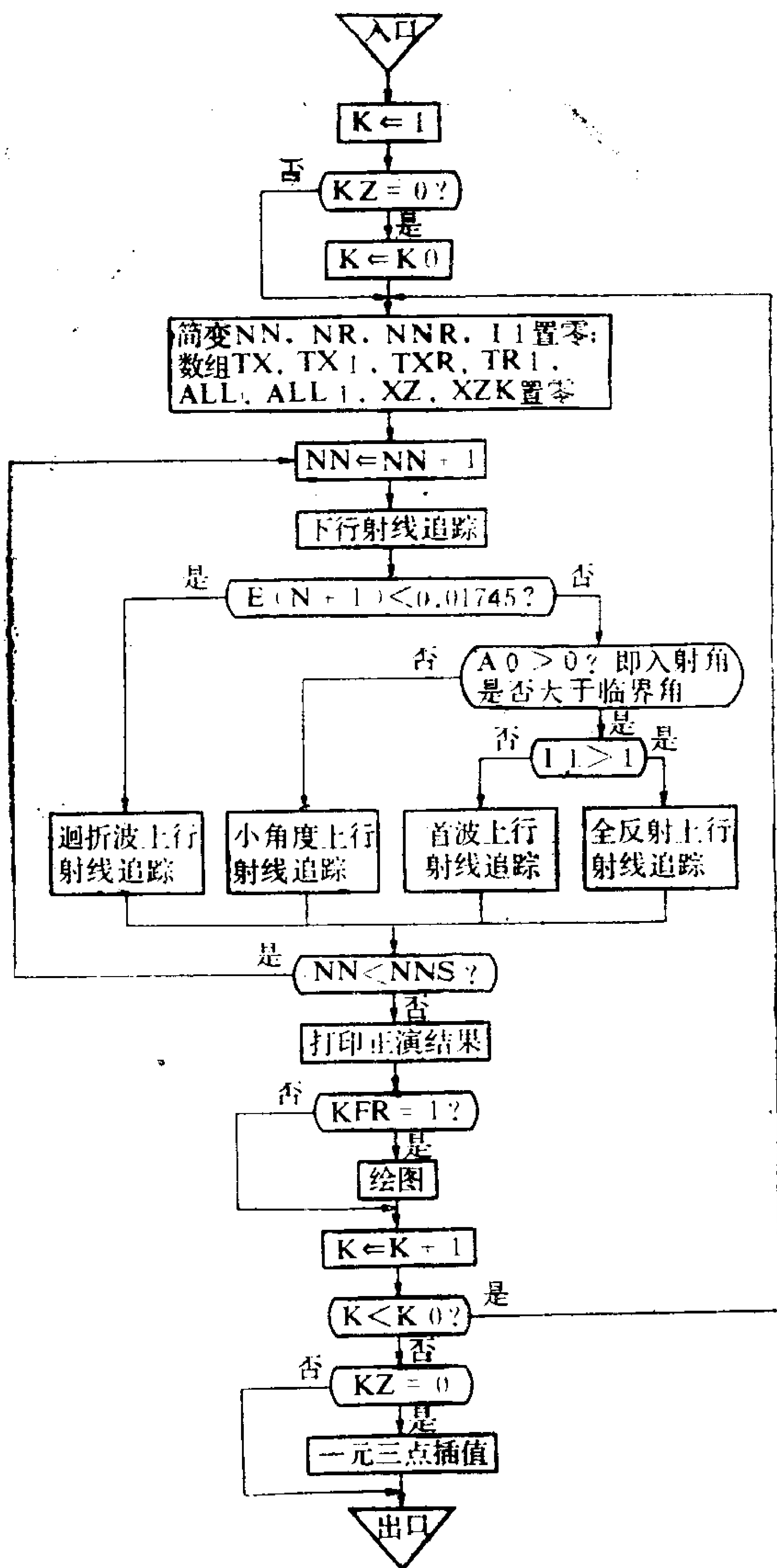



图10-5-6 ZYFT子程序框图

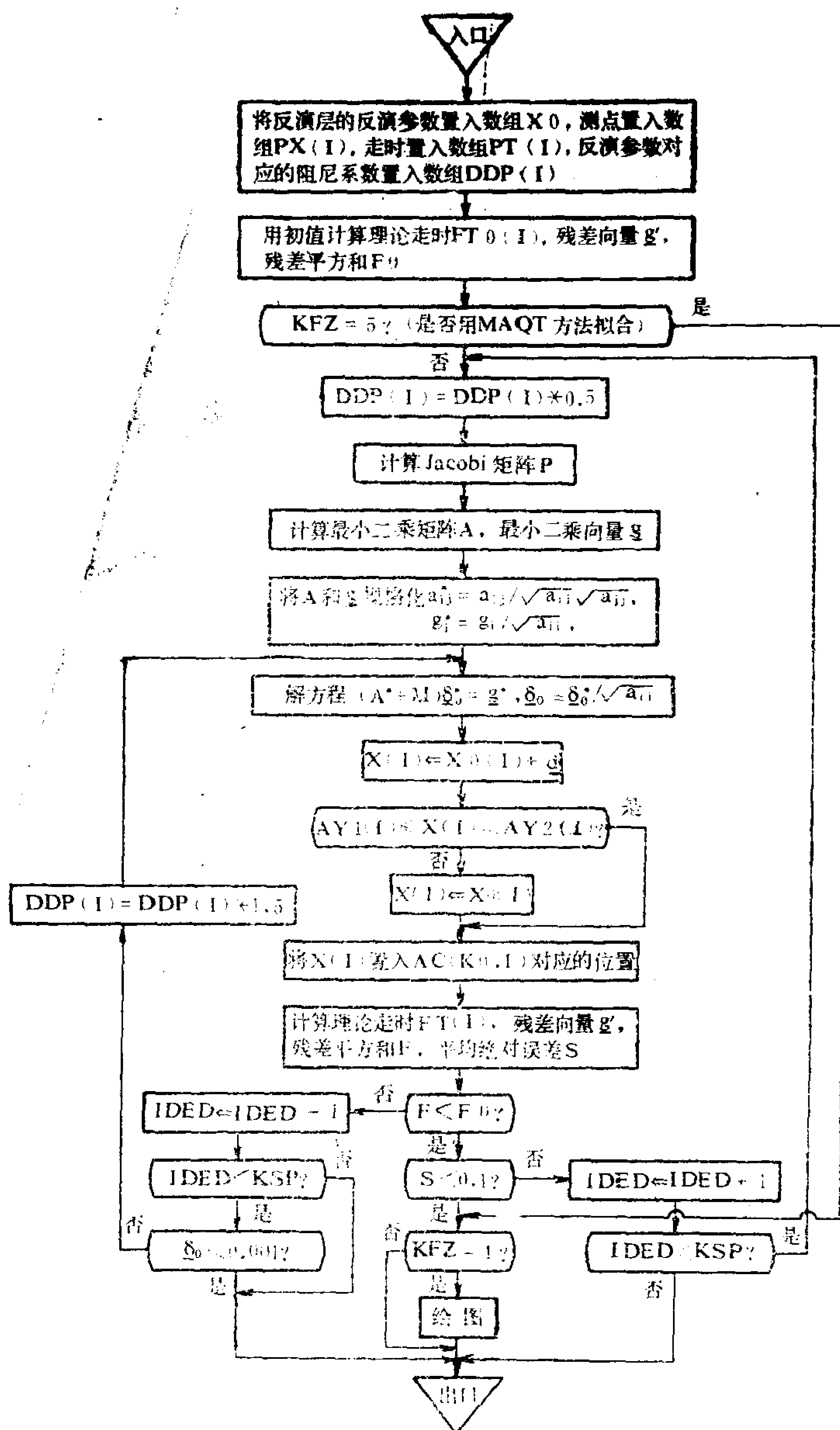



图10-5-7 MAQT子程序框图

## （五）输 出 标 志

＊＊＊YUAN SHI SHU JIU ＊＊＊——原始数据，将输入的数据全部输出。

＊＊＊＊＊——原始数据输出完毕标志。

＊＊1 LAYER＊＊——第一层。

LITTLE ANGLE REFLECTIVE-WAVE，NL 13——小角度反射波射线条数。

REFLECTIVE-WAVE——反射波时距数值。

REFRACTIVE-WAVE——折射波时距数值。

DIVING-WAVE——回折波时距数值。

＊＊＊END＊＊＊——计算结束标志。

＊＊＊ZHE SE BO＊＊＊——反演折射波标志。

＊＊＊FAN SE BO＊＊＊——反演反射波标志。

＊＊NO NLAYER＊＊——第 $N$ 层反演结果标志。

AC1——第 $N$ 层反演后的参数在绝对坐标系的值。

PX——测点炮检距。

PT——实测走时。

PTO——理论走时。

DPT——实测走时与理论走时之差。

DEDI——迭代次数。

AS——平均相对误差。

MAX——DPT 的最大值点号。

BS——DPT 的最大值。

＊＊＊＊ZONG JE KO＊＊＊＊——反演总结果。

＊＊＊END＊＊＊——计算结束标志。

如果 KI＝5，还输出下列标志：

CHE LIAN——实测走时值。

AC——变换到炮点为坐标系原点后的参数。

**FTO**——初始理论走时。

**XO**——参加反演的参数值。

**CAJ**——Jacobi 矩阵。

**AZE**——增广矩阵。

**L-AZE**——回代前的增广矩阵。

**AW**——改正量。

**X**——改正后的参数值。

**FTO*2***——模型参数加上改正向量后计算的理论走时。

**S**——平均绝对误差。

## （六）源 程 序

该程序是用 FORTRAN-IV 编写的，在 SPERRY UNIVAC 1100/10 计算机上进行了试算，得到了初步结果。

```
C        MAINPROGRAM
C
         DIMENSION FT(100), FTO(100), PX(100),PT(100),
            DPT(100), DDP(50),
            R1(50),XO(50),X(50),AZE(50, 50), CAJ(50, 50),
            AW(50)
         COMMON  /KF/  KFR, KFT, KFZ
         COMMON  /KZ1/  KC, KX, KY, KD, KJ
         COMMON  /TN/  TAPE, NNS
         COMMON  /EN/  LG, LF, LR
         COMMON  /ID/  KSP, IFL
         COMMON  /IHFG/  IHG, IHF, ITG, ITF
         COMMON  /FAN1/  M, MZG, MZF, IC, IB, MAX
         COMMON  /ZAN1/  AN,AO, Q, DT, DV, GA, A,
            RO, R, SS
         COMMON  /ANG1/  ALP
         COMMON  /ZAN/MTAN, MM, SH1, SH2, N, NN,
           IUP, IO, KI, KZ, MM1, MAC
```

```
      COMMON/FAN/KK, K0, KO, LN
      COMMON/FIG/GLX, LX, GLY, LY, XMINR, XM-
         AXR, YMINR, YMAXR
      COMMON/FIG1/ GLXT, LXT,GLYT, LYT, XMINT,
         XMAXT, YMINT, YMAXT
C     COMMON /ZAC/ AC1(10, 50), AC(10,50), AO(10,
         50), IX(11, 51)
      COMMON /XZT/ XZ(500, 3), XZO(500, 3), XZK
         (50, 3), E(500)
      COMMON /FXF/ TX(200, 2), TX1(200, 2), ALL
         (200, 2))ALL1(200, 2),
    1 TXR(200, 2), TXR1(200, 2)
      COMMON /ANG/ B11(500), B12(500), B21(50)
      COMMON /ST/ IH(500),PGJ(100),GST(500), FST
         (500)
      COMMON/FY/AY1(5, 21), AY2(5, 21), DDO(21),
         XDJ(21)
      COMMON/ZL/XZL(3)
      WRITE(6, 1)
      1FORMAT(/////, 50X, 3H* * *, 4HYUAN, 2X,
         3HSHI, 2X, 3HSHU, 2X, 3HJIU, 3H*** )
      READ(5, *) MTAN, MG, MF, M, MZ, IO, ME,
         KC, KX, KY, KD, KJ, KI, KZ,
      1KFR, KFT, KFZ, KSP
      WRITE(6,   3)MTAN, MG, MF, M, MZ, IO, ME,
         KC, KX, KY, KD, KJ, KI, KZ,
      1KFR, KFT, KFZ, KSP
      3 FORMAT(//, 5X, 4HMTAN, I2, 2X, 2HMG, I2,
         2X, 2HMF, I2, 2X, 1HM, I3, 2X, 2HMZ, I3,
         2X, 2HIOI2, 2X, 2HME, I2, 2X, 2HKC, I2,
         X, 2HKX, I2, 2X, 2HKY, I2, 2X, 2HKD, I2,
         2X, 2HKJ, I2, 2X, 2HKI, I2, 2X, 2HKZ, I2,
```

```
      2X,
      3HKFR, I2, 2X, 3HKFT, I2, 2X, 3HXFZ, I2,
      2X, 3HKSP, I2)
   READ(5, •)  TAPE, NNS
   WRITE(6, 2)  TAPE, NNS
   2FORMAT(//, 5X, 4HTAPE, F6.3, 2X, 3HNNS, I2)
   READ(5, •)  GLX, LX, GLY, LY, XMINR,  XMA
    XR, YMINR, YMAXR
   WRITE(6,     4)  GLX, LX, GLY, LY, XMINR,
      XMAXR, YMINR, YMAXR
 4 FORMAT(//, 5X, 3HGLX, F8.3, 2X, 2HLX, I2,
      2X, 3HGLY, F8.3, 2X, 2HLY, I2, 2X
      5HXMINR, F8. 3, 2X, 5HXMAXR, F8. 3, 2X,
      5HYMINR. F8. 3, 2X, 5HYMAXR, F8. 3)
   READ(5, •)  GLXT, LXT, GLYT, LYT,XMINT, X-
       MAXT, YMINT, YMAXT
   WRITE(6, 5)  GLXT, LXT, GLYT, LYT, XMINT ,
      XMAXT, YMINT, YMAXT
 5 FORMAT(//, 5X, 4HGLXT, F7.3, 2X, 3HLXT,
      I2, 2X, 4HGLYT, F7.3, 2X, 3HLYT, I2,
      2X, 5HXMINT, F7.3,2X, 5HXMAXT, F7.3, 2X,
      5HYMINT, F7.3, 2X, 5HYMAXT, F7.3)
10 MM=MG+MF
   MM1=MM+1
   MAC=3•MTAN +
   MAX=MAC+
   MF1=MG+1
   MD=MZ+MM
   IHG=0
   IHF=0
   ITG=0
   ITF=0
```

```
      DO    17 I0=1, MM1
      DO    17 J0=1, MAC
      AC1(I0, J0)=0.0
      AC    (I0, J0)=0.0
 17   A0 (I0, J0)=0.0
      DO    19 I0=1, MM1
      DO    19 J0=1, MAX
 19   IX (I0, J0)=0.0
      CALL PLOTS(0.0,0.0,9)
      CALL PLOT(0.5,0.5,-3)
 28   IF(IO.NE.0)GO TO 33)
     READ(5, •)  ((AC(I, J), J=1, MAC), I=1, MM1)
      WRITE(6, 31)
 31   FORMAT(//, 5X, 2HAC  )
      DO 32 I=1, MM1
 32   WRITE(6, 23)    (AC(I, J), J=1, MAC)
      GO TO 44
 33   READ(5,    •  )  SH1, SH2
      WRITE(6,     7)  SH1, SH2
  7   FORMAT(//, 5X, 3HSH1, F10.3, 5X, 3HSH2, F10.3)
      READ(5, •)((AC1(I, J), J=1, MAC), I=1, MM1)
      WRITE(6, 35)
 35   FORMAT(//, 5X, 3HAC1)
      DO 34 I=1, MM1
 34   WRITE(6, 23)    (AC1(I, J), J=1, MAC)
      DO 211 I=1, MM1
      DO 211 J=1, 5
211   AC(I, J)=AC1(I, J)
      DO 215 I=1, MM1
      AC(I, 6)=AC1(I, 6)+AC1(I, 7)•SH1
      AC(I, 7)=AC1(I, 7)
      DO 215 J=1, MTAN
```

```
      AC(I, 3*J+5)=AC1(I, 3*J+5)
      IF(AC1(I, 3*J+5).EQ.0.0.) GO TO 213
      AC(I, 3*J+6)=AC1(I, 3*J+6)-SH1
      GO TO 215
213   AC(I,3*J+6)=AC1(I, 3*J+6)
215   AC(I, 3*J+7)=AC1(I, 3*J+7)
      IF(SH2.GE.0.0) GO TO 44
      GLX=-GLX
      XMINR=-XMINR
      XMAXR=-XMAXR
      GLXT=-GLXT
      XMINT=-XMINT
      XMAXT=-XMAXT
 44   DO 61 I=1, MM1
      DO 61 J=1, MAC
 61   A0(I, J)=AC(I, J)
      IF(KZ.NE.1) GO TO 12
      KO=MM
      K0=MM1
      LG=1
      LF=1
      LR=1
      CALL ZYFT(FTO, PX, 20)
      GO TO 73
 12   READ(5,   *)  (IH(I), I=1, MD)
      WRITE(6,     9)
  9   FORMAT(//, 5X, 2HIH  )
      WRITE(6, 11)  (IH(I), I=1, MD)
 11   FORMAT(20I6)
      MZG=0
      MZF=0
      IF(ME.EQ.0)  GO TO 14
```

```
      DO 13 I=1, MG
13    MZG=MZG+IH(I)
14    IF(ME.LT.0)GO TO 16
      DO    15 I=MF1, MM
15    MZF=MZF+IH(I)
16    IHG=MM
      IHF=MM+MZG
      READ(5,  *  )  (PGJ(I), I=1, M)
      WRITE(6, 21)
21    FORMAT(//, 5X, 3HPGJ  )
      WRITE(6, 23)  (PGJ(I), I=1, M)
23    FORMAT(12F10.3)
      READ(5,  *)((IX(I, J), J=1, MAX), I=1, MM1)
      WRITE(6, 38)
38    FORMAT(//, 5X, 2HIX)
40    FORMAT(24I5)
      DO 80 I=1, MM1
80    WRITE(6, 40)    (IX(I, J), J=1, MAX)
      IF(KQY.E.1)  GO TO 46
      DO 39 I=1, MM1
      AY1(I, 1)=2.0
      AY1(I, 2)=2.0
      AY1(I, 3)=2.0
      AY1(I, 4)=0.00001
      AY1(I, 5)=0.00001
      AY1(I, 6)=1.0
      AY1(I, 7)=0.01745
      AY2(I, 1)=10.0
      AY2(I, 2)=10.0
      AY2(I, 3)=10.0
      AY2(I, 4)=1.0
      AY2(I, 5)=1.0
```

```
      AY2(I, 6)=40.0
      AY2(I, 7)=0.5
      DO 39 J=1, MTAN
      AY1(I, 3*J+5)=0.1
      AY1(I, 3*J+6)=-500.0
      AY1(I, 3*J+7)=0.1
      AY2(I, 3*J+5)=10.0
      AY2(I, 3*J+6)=500.0
   39 AY2(I, 3*J+7)=50.0
      GO TO 45
   46 READ(5, *) ((AY1(I, J), J=1, MAC), I=1, MM1)
      WRITE(6, 41)
   41 FORMAT(//, 5X, 3HAY1)
      DO 82 I=1, MM1
   82 WRITE(6, 23) (AY1(I, J), J=1, MAC)
      READ(5, *) ((AY2(I, J), J=1, MAC), I=1, MM1)
      WRITE(6, 43)
   43 FORMAT(//, 5X, 3HAY2)
      DO 84 I=1, MM1
   84 WRITE(6, 23) (AY2(I, J), J=1, MAC)
   45 IF(KD.EQ.1) GO TO 48
      D=1.0
      DDO(1)=D
      DDO(2)=D
      DDO(3)=D
      DDO(4)=D*10.0
      DDO(5)=D*20.0
      DDO(6)=D
      DDO(7)=D*10.0
      DO 47 J=1, MTAN
      DDO(3*J+5)=D*2.0
      DDO(3*J+6)=D*2.0
```

```
47  DDO(3 * J + 7) = D * 10.0
    GO TO 51
48  READ(5, *)  (DDO(I), I=1, MAC)
    WRITE(6, 49)
49  FORMAT(//, 5X, 3HDDO)
    WRITE(6, 23)  (DDO(I), I=1, MAC)
51  IF(KJ.EQ.1)  GO  TO  55
    XDJ(1) = 0.1
    XDJ(2) = 0.1
    XDJ(3) = 0.01
    XDJ(4) = 0.01
    XDJ(5) = 0.01
    XDJ(6) = 0.1
    XDJ(7) = 0.001
    DO 53 J = 1, MTAN
    XDJ(3 * J + 5) = 0.1
    XDJ(3 * J + 6) = 0.1
53  XDJ(3 * J + 7) = 0.1
    GO TO 64
55  READ(5, *)  (XDJ(I), I=1, MAC)
    WRITE(6, 57)
57  FORMAT(//, 5X, 3HJDX)
    WRITE(6, 23)  (XDJ(I), I=1, MAC)
C
64  FY=GLY+7.0
    CALL PLOT(1.0, FY, -3)
    IC=MM
    IF(ME.EQ.0)  GO TO 1010
    READ(5, *)  (GST(I), I=1, MZG)
    WRITE(6. 25)
25  FORMAT(//, 5X, 3HGST)
    WRITE(6, 23)  (GST(I), I=1, MZG)
```

```
      IF(ME.LT.0)  WRITE(6, 60)
   60 FORMAT(///, 60X, 5H* * * * *)
      WRITE(6, 63)
   63 FORMAT(////, 5X, 3H* * *, 3HZHE, 2X, 2HSE,
         2X, 2HBO, 3H* * *)
      LG=1
      IB=0
      DO 65 KO=1, MG
      K0=KO+1
      IM=IH(KO)
      IF(IM.EQ.0)GO TO 65
      NX=IX(KO, 1)
      NNX=NX+1
      CALL MAQT(FT, FTO, PX, PT, DPT, IM, DDP,
         R1, XO, X, AW. NX, AZE, NNX, CAJ)
   65 CONTINUE
 1009 IF(ME.GE.0)  AC(K0,  +1,1)=AC(K0, 3)
 1010 IF(ME.LT.0)  GO TO 68
      READ(5, *)    FST(I, I=1, MZF)
      WRITE(6, 27)
   27 FORMAT(//, 5X, 3HFST  )
      WRITE(6. 23)(FST(I), I=1, MZF)
      WRITE(6, 60)
      WRITE(6, 66)
   66 FORMAT(////, 54X, 3H* * *, 3HFAN. 2X, 2HSE,
         2X, 2HBO, 3H* * *)
      LF=1
      LG=0
      IB=0
      DO 67 KO=MF1, MM
      K0=KO+1
      IM=IH(MG+KO)
```

```
      IF(IM.EQ.0)GO TO 67
      NX=IX(K0, 1)
      NNX=NX+1
      CALL MAQT(FT, FTO, PX, PT, DPT, IM, DDP,
         R1,XO, X, AW, NX, AZE, NNX, CAJ)
 67   CONTINUE
 68   WRITE(6, 69)
 69   FORMAT(////, 54X, 4H****, 4HZONG, 1X,
         2HJE, 1X, 2HKO, 4H****/)
      IF(IO.NE.0)  GO TO 71
      DO 70 I=1, MM1
 70   WRITE(6, 23)   (AC(I, J), J=1, MAC)
      GO TO 81
 71   DO 231 I=1, MM1
      DO 231 J=1, 5
231   AC1(I, J)=AC(I, J)
      DO 235 I=1, MM1
      AC1(I, 6)=AC(I, 6)-AC(I, 7)*SH1
      AC1(I, 7)=AC(I, 7)
      DO  235  J=1, MTAN
      AC1(I, 3*J+5)=AC(I, 3*J+5)
      IF(AC(I, 3*J+5).EQ.0.0)  GO TO 233
      AC1(I, 3*J+6)=AC(I,3*J+6)+SH1
      GO TO 235
233   AC1(I,3*J+6)AC(I,3*J+6)
235   AC1(I,3*J+7)=AC(I,3*J+7)
      DO 72 I=1,MM1
 72   WRITE(6, 23)   (AC1(I,J), J=1,MAC)
 81   IF(KFZ.NE.1)GO TO 73
      CALL RAH(GLXT,LXT,GLYT,LYT, 0.2,1, XMINT,
         XMAXT, YMINT,YMAXT)
      CALL PLOT(0.,-5.0,-3)
```

地

11-

```
      CALL PLOT(0.0,-GLY,-3)
   79 CALL RAH(GLX,LX,GLY,LY,0.2, 2, XMINR, XM-
     AXR, YMINR, YMAXR)
      CALL FIGZ  (MM1)
      CALL PLOT (0.0, 0.0, 3)
   73 WRITE(6, 75)
   75 FORMAT(//, 60X, 9H* * *END* * *)
      CALL PLOT(0., 2.0, -3)
      CALL PLOT(0.0, 0.0, 999)
      STOP
      END
C     PLOTING THE SEISMIC RAYS ON A CHART
      SUBROUTINE FIGR
      COMMON /K1/ K
      COMMON /FIG/ GLX, LX, GLY, LY, XMINR, XM
     AXR, YMINR, YMAXR
      COMMON /ZAC/ AC1(10, 50), AC(10, 50), A0(10,
     50), IX(11, 51)
      COMMON /ZAN/ MTAN, MM, SH1, SH2, N, NN,
     IUP, IO, KI, KZ, MM1, MAC
      COMMON /XZT/ XZ(500, 3), XZO(500, 3), XZK
     (50, 3), E(500)
 2002 X1=XZ(1, 1)
      X1      =X1      *GLX/(XMAXR-XMINR)
      Y1=XZ(1, 2)
      Y1      =Y1      *GLY/(YMAXR-YMINR)
      Y1=Y1
      CALL PLOT(X1, Y1, 3)
      IF(A0(K+1, 2).NE.1.0) CALL SMOOT (X1, Y1,0)
      IF(A0(K+1, 2).EQ.1.0) CALL PLOT(X1, Y1, 2)
      N1=N+1
      DO 2001 I=1, N1
```

```
      XX=XZ(I, 1)
      XX=XX*GLX/(XMAXR-XMINR)
      IF(SH2.LT.0.0.AND.I.EQ.1)XX=-XX
      YY=XZ(I, 2)
      YY=YY*GLY/(YMAXR-YMINR)
      YY=-YY
      IF(SH2.LT.0.0.AND.XX.GT.0.0)  GO TO 2009
      IF(SH2.GE.0.0.AND.XX.LT.0.0)  GO TO 2009
      IF(ABS(XX).GT.ABS(GLX))  GO TO 2009
      IF(A0(K+1, 2).NE.1.0)  CALL SMOOT (XX, YY,
     -2)
2001  IF(A0(K+1, 2).EQ.1.0)  CALL PLOT(XX, YY, 2)
2009  CONTINUE
      IF(A0(K+1, 2).NE.1.0)  CALL SMOOT(XX, YY,
     -24)
      RETURN
      END
C     PLOTING THE SURFACE AND INTERFACES ON
C        A CHART
      SUBROUTINE FIGZ(M11)
      COMMON /FIG/ GLX, LX, GLY, LY, XMINR,
        XMAXR, YMINR, YMAXR
      COMMON /ZAN/ MTAN, MM, SH1, SH2, N,
        NN, IUP, ID, KI, KZ,MM1,MAC
      COMMON /ZAC/ AC1(10,50),AC(10,50),A0(10,50),
       IX(11,51)
      CALL NEWPEN(2)
      DO 203 I=1,M11
      IF(A0(I,7).NE.0.0)  GO TO 1001
203   CONTINUE
      IF(MTAN.NE.0)GO TO 1001
      DO 11    I=1,M11
```

```
      CALL PLOT(0.0,  0.0,  3)
      YM=GLY-AC    (I,6)*GLY/(YMAXR-YMINR)
      CALL PLOT(0.0,YM,3)
      CALL PLOT(GLX,YM,2)
      CALL PLOT(0.0,0.0,3)
      XS=0.2
      IF(SH2.LT.0.0)  XS=-3.0
      YS=GLY-AC(   I,6)*GLY/(YMAXR-YMINR)-0.3
      CALL PLOT(XS,YS,  3)
      CALL SYMBOL(XS,  YS,  0.2,  2HV=,    0.0,  3)
      CALL NUMBER(999.0,999.0,  0.2,  AC(I,2),0.0,  3)
      IF(I.EQ.1)  GO TO 11
  105 YSYS+0.4
      IF(A0(I,2).EQ.1.0.OR.A0(I,2).EQ.0.1)  GO TO 301
      CALL SYMBOL(XS,YS,0.2,2HV=,0.0,3)
      CALL NUMBER(999.0,  999.0,  0.2, AC( I,2), 0.0, 3)
  301 IF(I.EQ.1)  GO TO 11
      IF(A0(I,2).NE.0.1) GO TO 11
  107 YS=YS+0.6
      CALL SYMBOL(XS,YS,0.2,9HVGX=F(X)* ,0.0,9)
      CALL NUMBER(999.0,  999.0,  0.2,  AC( I,4),0.0,3)
      YS=YS+0.4
      CALL SYMBOL(XS,YS,0.2,9HVGZ=F(Z)* ,0.0,9)
      CALL NUMBER(999.0,  999.0,  0.2,AC( I,5),0.0,3)
      CALL PLOT(0.,0.,3)
   11 CONTINUE
      GO TO 1007
 1001 FX=0.5
      JX=(XMAXR-XMINR)/FX
      DO 103 I=1,M11
      DO 101 J=1, JX
      XX=FX*FLOAT(J-1)
```

```
      YY=Z(XX,I)
      XX=XX*GLX/(XMAXR-XMINR)
      YY=YY*GLY/(YMAXR-YMINR)
      IF(SH2.LT.0.0.AND.I.EQ.1)  XX=-XX
      YY=GLY-YY
      IF(J.EQ.1)  YS=YY-0.3
      IF(J.EQ.1)  CALL SMOOT(XX,YY,0)
 101  IF(J.NE.1)  CALL SMOOT  (XX,YY,-2)
       CALL SMOOT(XX,YY,-24)
      CALL PLOT(0.0,  0.0,  3)
      XS=0.2
      IF(SH2.LT.0.0)  XS=-3.0
      CALL PLOT(XS,YS,  3)
       CALL SYMBL(XS,YS,0.2,2HV=,0.0,3)
      CALL NUMBER(999.0,999.,0.2,AC(I,3),0.0,3)
      IF(I.EQ.1)  GO TO 103
 109  YS=YS+0.4
      IF(A0(I,2).EQ.1.0.OR.A0(I,2).EQ.0.1) GO TO 303
      CALL SYMBOL(XS,YS,0.2,2HV=,0.0,3)
      CALL NUMBER(999.0,  999.0,0.2,AC(  I,2),0.0,  3)
 303  IF(I.EQ.1)GO TO 103
      IF(A0(I,2).NE.0.1)GO TO 103
1005  YS=YS+0.6
      CALL SYMBOL(XS,YS, 0.2, 9HVGX=F(X) * ,0.0,9)
      CALL NUMBER(999.0,  999.0,  0.2,AC( I,4),0.0,  3
      YS=YS+0.4
      CALL SYMBOL(XS,YS,0.2,9HVGZ=F(Z) * ,0.0,9)
      CALL NUBER(999.0,  999.0,  0.2,  AC(  I,5),0.0,  3)
 103  CONTINUE
1007  CALL PLOT(0.0,  0.0,  3)
      CALL NEWPEN(1)
      RETURN
```

```
      END
C     PLOTING THE TRAVEL—TIME CURVES OF THE
      REFLECTED OR
C     REFRACTED WAVE ON A CHART
      SUBROUTINE FIGT(TX,NNF,K)
      DIMENSION TX(200,2)
      COMMON/ZAN/MTAN,MM,SH1,SH2,N, NN, IUP,
         IO,KI,KZ,MM1,MAC
      COMMON  /FIG1/  GLXT, LXT,GLYT, LYT, XMI
         NT, XMAXT,YMINT,YMAXT
      J = 0
      DO 101 I=1,NNF
      XX = TX(I,1)
      YY = TX(I,2)
      XX = XX * GLXT/(XMAXT - XMINT)
      IF(SH2.LT.0.0.AND.I.EQ.1)  XX = - XX
      YY = YY * GLYT/(YMAXT - XMINT)
      IF(ABS(XX).GT.ABS(GLXT))  GO TO 101
      IF(SH2.GE.0.0AND.XX.LT.0.0)  GO.TO  101
      IF(SH2.LT.0.0.AND.XX.GT.0.0)  GO TO 101
      IF(YY.GT.GLYT.OR.YY.LT.0.0)  GO TO 101
      J = J + 1
      IF(J. EQ. 1)  CALL SMOOT(XX,YY,0)
      IF(J.NE.1)  CALL SMOOT(XX,YY,-2)
  101 CONTINUE
      IF(XX.GE.GLXT.AND.SH2.GE.0.0)XX = GLXT - 1.0
      IF(XX.LT.GLXT.AND.SH2.LT.0.0) XX = GLXT + 1.0
      IF(YY.GE.GLYT)  YY = GLYT - 1.0
      IF(YY.LT.0.0)  YY = 1.0
      CALL SMOOT(XX,YY,-24)
      XX = XX - 0.3
      YY = YY - 0.1
```

```
      FK=K
      CALL NUMBER(XX,YY,0.3,FK,0.0,-1)
      CALL PLOT(0.0, 0.0, 3)
      RETURN
      END
C     PLOTING THE TRAVEL—TIMECURVESOF DIVING
C     WAVE
      SUBROUTINE FIGTR(TX,NNF,K)
      DIMENSION TX(200,2)
      COMMON/FIG1/GLXT,LXT,GLYTLYT,XMINT,XM-
         AXT, YMINT,YMINT,YMAXT
      COMMON/ZAN/MTAN,MM,SH1,SH2,N,NN, IUP,
         IO,KI,KZ,MM1,MAC
      DO 101 J=1,NNF
      I=NNF+1-J
      XX=TX(I,1)
      YY=TX(I,2)
      XX=XX*GLXT/(XMAXT-XMINT)
      IF(SH2.LT.0.0.AND.I.EQ.NNF)  XX=-XX
      YY=YY*GLYT/(YMAXT-XMINT)
      IF(ABS(XX).GT.ABS(GLXT))  GO TO 101
      IF(SH2.LT.0.0.AND.XX.GT.0.0)  GO TO 101
      IF(SH2.GE.0.0.AND.XX.LT.0.0)  GO TO 101
      IF(YY.GT.GLYT.OR.YY.LT.0.0)  GO TO 101
      IF(I.EQ.NNF)CALL SMOOT(XX,YY,0)
      IF(I.NE.NNF) CALL SMOOT(XX,YY,-2)
  101 CONTINUE
      IF(XX.GE.GLXT.AND.SH2.GE.0.0)XX=GLXT-1.0
      IF(XX.LT.GLXT.AND.SH2.LT.0.0)XX=GLXT+1.0
      IF(YY.GE.GLYT)YY=GLYT-1.0
      IF(YY.LT.0.0)   YY=1.0
      CALL SMOOT(XX,YY,-24)
```

```
      XX=XX-0.3
      YY=YY-0.1
      FK=K
      CALL NUMBER(XX,YY,0.3,FK,0.0,-1)
      CALL SYMBOL(999.0,999.0,0.3,1HD,0.0,1)
      CALL PLOT(0.0, 0.0, 3)
      RETURN
      END
      PLOTING THE TRAVEL—TIME CURVES FITTING
C     FOR OBSERVATION AND
C     COMPUTATION IN INVERSION
      SUBROUTINE FIGT1(PX,PT,NNF, K)
      DIMENSION PX(200), PT(200)
      COMMON /ZAN/ MTAN, MM, SH1, SH2, N, NN,
         IUP, IO, KI, KZ, MM1, MAC
      COMMON /FIG1/ GLXT,LXT,GLYT, LYT, XMI
         NT, XMAXT, YMINT, YMAXT
      X1=PX(1)
      X1=X1*GLXT/(XMAXT-XMINT)
      Y1=PT(1)
      Y1=Y1*GLYT/(YMAXT-YMINT)
      CALL SMOOT(X1, Y1, 0)
      DO 101 I=1, NNF
      XX=PX(I)
      YY=PT(I)
      XX=XX*GLXT/(XMAXT-XMINT)
      IF(SH2.LT.0.0.AND.I.EQ.1) XX=-XX
      YY=YY*GLYT/(YMAXT-YMINT)
      IF(ABS(XX).GT.ABS(GLXT)) GO TO 101
      IF(SH2.GE.0.0.AND.XX.LT.0.0) GO TO 101
      IF(SH2.LT.0.0.AND.XX.GT.0.0) GO TO 101
      IF(YY.GT..GLYT.OR.YY.LT.0.0) GO TO 101
```

```
      CALL SMOOT(XX, YY, -2)
 101  CONTINUE
      IF(XX.LT.GLXT.AND.SH2.LT.0.0)  XX=GLXT
     +1.0
      IF(XX.GE.GLXT.AND.SH2.GE.0.0)  XX=GLXT
     -1.0
      IF(YY.GE.GLYT)  YY=GLYT-1.0
      IF(YY.LT.0.0)   YY=1.0
      CALL SMOOT(XX, YY, -24)
      YY=YY-0.1
      XX=XX-0.3
      F=K
      CALL NUMBER(XX, YY, 0.3, F, 0.0, -1)
      CALL PLOT(0.0, 0.0, 3)
      RETURN
      END
C     PLOTING THE COORDINATE SYSTEM
      SUBROUTINE RAH(GLX,IX,GLY, IY, GH, KOM,
     XMIN, XMAX, YMIN, YMAX)
      COMMON /ZAN/ MTAN, MM, SH1,SH2,N, NN,
     IUP, IO, KI, KZ, MM1, MAC
      COMMON /KF/ KFR, KFT, KFZ
      COMMON /KZ1/ KC, KX, KY, KD, KJ
      ANG=90.0
      GIX=IX
      GIY=IY
      A=GLX/GIX
      B=GLY/GIY
      GG=-GH
      IF(SH2.LT.0.0)  GG=-GG
      DO 210 I=1, IX
      GI=I
```

```
      C = GI * A
      CALL PLOT(C, 0., 2)
      CALL PLOT(C, GG, 2)
210   CALL PLOT(C, 0., 2)
      DO 220 I = 1, IY
      GI = I
      D = GI * B
      CALL PLOT(C, D, 2)
      CALL PLOT(C - GG, D, 2)
220   CALL PLOT(C, D, 2)
      DO 230  I = 1, IX
      GI = IX - I
      E = GI * A
      CALL PLOT(E, D, 2)
      CALL PLOT(E, D - GG, 2)
230   CALL PLOT(E, D, 2)
      DO 240  I = 1, IY
      GI = IY - I
      F = GI * B
      CALL PLOT(0., F, 2)
      CALL PLOT(GG, F, 2)
240   CALL PLOT(0., F, 2)
      CALL PLOT(0., F, 3)
      IDX = 5.0*(XMAX - XMIN)/GIX + 0.5
      IF(SH2.LT.0.0)  IDX = 5.0 * (XMAX - XMIN)/GIX
        - 0.5
      DX = IDX
      X0 = XMIN
      IXX = IX/5 + 1
      DO 250 I = 1, IXX
      GI = I - 1
      X = GI * A * 5.0 - 0.3
```

```
      Y=-0.7
      IF(KOM.NE.1)  X=GI*A*5 0-0.35
      IF(KOM NE.1)  Y=GLY+0.4
      CALL NUMBER  (X, Y, 0.3, X0, 0., -1)
250   X0=X0+DX
      IYY=IY/5+1
      IDY=5.0*(YMAX-YMIN)/GIY+0.5
      DY=IDY
      IF(KOM.NE.1)  DY=-DY
      Y0=YMIN
      IF(KOM.NE.1)  Y0=YMAX
      DO 260 I=1, IYY
      GI=I-1
      Y=GI*B*5.0-0.15
      X=-0.8
      IF(SH2.LT.0.0)  X=-X
      CALL NUMBER  (X, Y, 0.3, Y0, 0.0, -1)
260   Y0=Y0+DY
      CALL PLOT(0., 0., 3)
      IF(KOM.NE.1)  GO TO 300
      X=-1.1
      IF(SH2.LT.0.0)  X=-X
      Y=GLY-3.5
      IF(KC.EQ.1)  CALL SYMBOL(X, Y, 0.4, 11HT=R/6
         (SEC.), ANG, 11)
      IF(KC.NE.1)CALL SYMBOL(X, Y, 0.4, 6HT(SEC),
         ANG, 6
      X=GLX-3.0
      Y=-1.3
      CALL SYMBOL(X, Y, 0.4, 5HX(KM), 0., 5)
      X=GLX/2.0-4.9
      Y=-2.0
```

```
      CALL SYMBOL(X, Y, 0.5, 20HTIME-DISTANCE
         CURVE, 0., 20)
      GO TO 400
  300 X=GLX-3.0
      IF(SH2.LT.0.0)  X=GLX+3.0
      Y=GLY+0.7
      CALL SYMBOL(X, Y, 0.4, 5HX(KM), 0.0, 5)
      X=-1.1
      IF(SH2.LT.0.0)  X=-X
      Y=1.0
      CALL SYMBOL(X, Y,0.4,5HZ(KM),ANG,5)
  261 X=GLX/2.0-2.0
      Y=-1.5
      IF(KFZ.EQ.1)  GO TO 263
      CALL SYMBOL(X,Y,0.5,7HRAY FIG,0.0,7)
      GO TO 400
  263 X=GLX/2.0-4.0
      CALL SYMBOL(X,Y, 0.5, 15HCONVERSE RESULT,
         0.0, 15)
  400 CONTINUE
      CALL PLOT(0.0,0.0,3)
      RETURN
      END
      TRAVEL—TIME IN EXPLOSIVE COORDINATE
C     SYSTEM IS CHANGED INTO
C     TRAVEL—TIME IN OBSOLUTE COORDINATE SYS-
C     TEM
      SUBROUTINE MOVE(N1,TX,TX1)
      DIMENSION TX(200,2).TX1(200,2)
      COMMON /KZ1/ KC,KX,KY,KD,KJ
      COMMON /ZAN/ MTAN,MM,SH1,SH2,N,NN,
         IUP, IO,KI,KZ,MM1,MAC
```

```
      IF(IO.EQ.0)  GO TO 201
      TX1(N1,1) = SH1 + SH2*TX(N1,1)
      TX1(N1,2) = TX(N1,2)
      IF(KC.EQ.1)  TX1(N1,2) = TX1(N1,2) - TX1(N1,1)/KX
      GO TO 203
  201 IF(KC.EQ.1)  TX  (N1,2) = TX (N1,2) - TX (N1,1)/KX
  203 CONTINUE
      RETURN
      END
C
C     COMPUTING THE VERTICAL COORDINATE OF
C     INTERFACES
      FUNCTION Z(X,K)
      COMMON/ZAC/AC1(10,50),AC(10,50), A0(10, 50), IX
     (11,51)
      COMMON  /ZAN/  MTAN,MM,SH1,SH2,N, NN,
     IUP, IO,KI,KZ,MM1,MAC
      IF(IO.NE.0)X = X * SH2
      S = AC(K,6) + AC(K,7) * X
      D = 0.0
      DO 251  I = 1,MTAN
  251 D = D + AC(K,3 * I + 5) * (0.5 + ATAN((X - AC(K,3 * I
     + 6))/(AC(K,3 *  + 7) + 0.001))/3.141593)
      Z = D + S
      RETURN
      END
C
C     COORDINATE AND TRAVEL—TIME ABOUT THE
C     END OF  SEISMIC RAY IS
C     COMPUTED DURING THE RAY GOES FORWARD
C     WITH AN ANGLE STEP
      SUBROUTINE XY(C)
```

```
      COMMON  /XZT/  XZ(500,3), XZO(500, 3), XZK
     *  (50,3), E(500)
      COMMON  /ZAN/  MTAN,MM,SH1,SH2,N, NN,
     *  IUP, IO,KI,KZ,MM1,MAC
      COMMON/ZAN1/AN,AO,Q,DT,DV,GA,A,R0,R,SS
      IF(KI.EQ.1) WRITE(6,262)E(N),C,A,B
  262 FORMAT(5X,6HSEARCH,4HE(N),F6.3,1HC,F6.3,
     *  1HA,F6.3,1HB,F6.3)
      XX=R0*SIN(C)
      ZZ=R0*(1-COS(C))
      C5=COS(E(N))
      C6=SIN(E(N))
      IF(IUP.NE.1)  GO TO 261
      XZ(N+1,1)=XZ(N,1)+XX*C5-ZZ*C6
      XZ(N+1,2)=XZ(N,2)-XX*C6-ZZ*C5
      GO TO 2601
  261 XZ(N+1)=XZ(N,1)+XX*C5+ZZ*C6
      XZ(N+1,2)=XZ(N,2)+XX*C9-ZZ*C5
 2601 IF(IUP.EQ.1)  B=A-C
      IF(IUP.NE.1)  B=A+C
      C1=COS(A)
      C2=COS(B)
      C3=ALOG((1.0+C1)/(1.0-C1))*0.5
      C4=ALOG((1.0+C2)/(1.0-C2))*0.5
      DT=ABS((C3-C4)/GA)
      XZ(N+1,3)=XZ(N,3)+DT
      RETURN
      END
C
C     INTERSECTION POINT OF SEISMIC RAY FOR
C     INTERFACE IS SEARCHED
C     IN DIVIDE EQUALLY
```

```
      SUBROUTINE  SEARC(S1,S2)
      REAL MI
      COMMON  /K1/  K
      COMMON/ZAN1/AN,AO,Q,DT,DV,GA,A,R0, R,SS
      COMMON/ZAC/AC1(10,50),AC(10,50), A0(10, 50), IX
     (11,51)
      COMMON /ZAN/ MTAN,MM,SH1,SH2,N,NN,IUP,
     IO,KI,KZ,MM1,MAC
      COMMON /XZT/ XZ(500,3),XZO(500,3),XZK(50,3),E
     (500)
      IF(KI.EQ.1)  WRITE(6,3)
    3 FORMAT(5X,5HSEARC)
      T=0.0
      MS=0
      SI=S2
 2801 MI=(S2-S1)*0.5
 2802 MS=MS+1
      IF(MS.LT.100)  GO TO 280
      WRITE(6,279)
  279 FORMAT(//,5X,15H**SEARCH FAIL**)
      STOP
  280 IF(DV.NE.0.0.OR.A.NE.0.0)  GO TO 282
      D=COS(E N))
      S=SIN(E(N))
      IF(IUP.NE.1)  GO TO 281
      XZ(N+1,1)=XZ(N,1)+(T+MI)*D
      XZ(N+1,2)=XZ(N,2)-(T+MI)*S
      Y0=Z(XZ(N+1,1),K)
      GO TO 2803
  281 XZ(N+1,1)=XZ(N,1)+(T+MI)*D
      XZ(N+1,2)=XZ(N,2)+(T+MI)*S
 6003 Y0=Z(XZ(N+1,1)K+1)
```

```
      GO TO 2803
 282  SE=T+MI
      CALL XY(SE)
      IF(IUP.EQ.1)  Y0=Z(XZ(N+1,1),K)
      IF(IUP.NE.1)  Y0=Z(XZ(N+1,1),K+1)
2803  IF(ABS(XZ(N+1,2)-Y0).LE.0.01)  GO TO 2804
      IF(IUP.NE.1)  GO TO 287
      IF(XZ(N+1,2).LT.Y0)  GO TO 285
      T=T+MI
      S1=T
      S2=SI
      GO TO 2801
 285  SI=T+MI
      MI=MI*0.5
      GO TO 2802
 287  IF(XZ(N+1,2).LT.Y0)  GO  TO 289
      SI=MI+T
      MI=MI*0.5
      GO TO 2802
 289  T=T+MI
      S1=T
      S2=SI
      IF(KI.EQ.1)WRITE(6,201)  XZ(N+1,1),XZ(N+1,2),
         Y0,SS,K
 201  FORMAT(12F10.4)
      GO TO 2801
2804  SS=T+MI
      RETURN
      END
C
C     COMPUTING THE ANGLES
      SUBROUTINE ANGLE(K)
```

```
      COMMON  /ZAN/  MTAN,MM,SH1,SH2,N,NN,
     IUP,IO,KI,KZ,MM1,MAC
      COMMON/ZAN1/AN,AO,Q,DT,DV,GA,A,R0,R,SS
      COMMON/ANG1/ALP
      COMMON/ZAC/AC1(10,50),AC(10,50),A0(10,50),IX
     (11,51)
      COMMON/XZT/XZ(500,3),XZO(500,3),XZK(50,3),E
     (500)
      COMMON  /ANG/  B11(500),B12(500),B21(50)
      IF(KI.EQ.1)  WRITE(6,1)
      1FORMAT(5X,5HANGLE)
      IF(MTAN.NE.0)  GO TO 211
      BED=AC(K+1,7)
      IF(IUP.EQ.1)  BED=AC(K,7)
      GO TO 215
211   BS=AC(K+1,10)
      IF(IUP.EQ.1)  BS=AC(K,10)
      JC=1
      DO 213 I=2,MTAN
      DP=AC(K+1,3*I+7)
      IF(IUP.EQ.1)  DP=AC(K,3*I+7)
      IF(DP.GT.BS)  GO TO 213
      BS=AC(K+1,3*I+7)
      IF(IUP.EQ.1)  BS=AC(K,3*I+7)
      JC=I
213   CONTINUE
      BS=AC(K+1,3*JC+7)
      IF(IUP.EQ.1)  BS=AC(K,3*JC+7)
      DBS=BS/10.0
      IF(IUP.EQ.1)  GO TO 301
      AS1=XZO(NN,1)+DBS
      AS2=XZO(NN,1)-DBS
```

```
      ZSK1 = Z(AS1,K+1
      ZSK2 = Z(AS2,K+1)
      AD = ZSK1 - ZSK2
      GO TO 303
301   AD1 = XZK(K,1) + DBS
      AD2 = XZK(K,1) - DBS
      ZDK1 = Z(AD1,K)
      ZDK2 = Z(AD2,K)
      AD = ZDK1 - ZDK2
303   AK3 = AD/(DBS * 2.0)
      BED = ATAN(AK3)
215   AKN = 1.57079633
      B11(NN) = E(N+1-BED
      A11 = AKN - B11(NN)
      IF(IUP.NE.1)  GO TO 309
      IF(A0(K,2).NE.1)  GO TO 305
      IF(AC(K,1).LE.AC.(K,3))  GO TO 304
      ALP = ARSIN(AC(K,3)/AC(K,1))
      IF(A11.GE.ALP)  GO TO 3001
304   AG1 = AC(K,1) * SIN(A11)/AC(K,3)
      IF(AG1.GT.1.0)  AG1 = 1.0
      IF(AG1.LT.-1.0)  AG1 = -1.0
      A21 = ARSIN(AG1)
      B21  (K) = AKN - (A21+BED)
      GO TO 3001
305   IF(A0(K,2).NE.O.1)  GO TO 307
      Z1 = Z(XZ(N+1,1),K-1)
      AC(K,2) = AC(K,1) + SQRT((AC(K,4) * XZ(N+1,2))
          * * 2+ (AC(K,5) * )XZ(N+1,2) - Z1)) * * 2)
307   IF(AC(K,2).LQ.AC(K,3))GO TO 308
      ALP = ARSIN(AC(K,3)/AC(K,2))
      IF(A11.GE.ALP)  GO TO 3001
```

```
308  AG1 = AC(K,2) * SIN(A11)/AC(K,3)
     IF(AG1.GT.1.0)AG1 = 1.0
     IF(AG1.LT.-1.0)  AG1 = -1.0
     A21 = ARSIN(AG1)
     B21(K) = AKN - (A21 + BED)
     GO TO 3001
309  IF(A0(K+1,2).NE.1)  GO TO 311
     IF(AC(K+1,3).LE.AC(K+1,1))  GO TO 310
     ALP = ARSIN(AC(K+1,1)/AC(K+1,3))
     IF(A11.GE.ALP)  GO TO 3001
310  AG1 = AC(K+1,3) * SIN(A11)/AC(K+1,1)
     IF(AG1,GT.1.0)  AG1 = 1.0
     IF(AG1.LT.-1.0)  AG1 = -1.0
     A12 = ARSIN(AG1)
     B12(NN) = AKN - (A12 - BED)
     GO TO 3001
311  IF(A0(K+1,2).NE.0.1)  GO TO  313
     Z2 = Z(XZ(N+1,1),K)
     AC(K+1,2) = AC(K+1,1) + SQRT((AC(K+1,4) * XZ(N+
        1,2)) * * 2+ (AC(K+1,5) * (XZ(N1+1,2) - Z2)) * * 2
313  IF(AC(K+1,3).GT.AC(K+1,2))  ALP = ARSIN(AC(K
        +1,2)/AC(K+1,3))
     IF(A11.GE.ALP)GO TO 3001
     AG1 = AC(K+1,3) * SIN(A11)/AC(K+1,2)
     IF(AG1.GT.1.0)AG1 = 1.0
     IF(AG1.LT.-1.0)AG1 = -1.0
     A12 = ARSIN(AG1)
     B12(NN) = AKN - (A12 - BED)
3001 AO = A11 - ALP
     IF(KI.NE.2)GO TO 317
     WF = 180/3.141592654
     A11N = A11 * WF
```

```
      A12N = A12 * WF
      ALPN = ALP * WF
      EN = E(N+1) * WF
      BEDN = BED * WF
      WRITE(6,315) NN,IUP,N,A11N,A12N,ALPN,EN,BED
         N,XZ(N+1,1),XZ(N+1,2),1XZ(N+1,3)
  315 FORMAT(2X,3I4,5F9.2,3F10.3)
  317 CONTINUE
      RETURN
      END

C     SEISMIC RAYS IS TRACED IN K—THLAYER
      SUBROUTINE TRACE(K)
      COMMON/OUT/OUT1,OUT2
      COMMON/FIG/GLX,LX,GLY,LY,XMINR,XMAXR,
         YMINR,YMAXR
      COMMON/ZAC/AC1(10,50),AC(10,50),A0(10,50),IX
         (11,51)
      COMMON/ID/KSP,IFL
      COMMON/ZAN/MTAN,MM,SH1,SH2,N,NN,IUP,
         IO,KI,KZ,MM1,MAC
      COMMON/KF/KFR,KFT,KFZ
      COMMON/ZAN1/AN,AO,Q,DT,DV,GA,A, R0,R, SS
      COMMON/XZT/XZ(500,3),XZO(500,3),XZK(50,3),E
         (500)
      COMMON/ANG/B11(500),B12(500),B21(50)
 3501 OUT1 = 0.0
      OUT2 = 0.0
      R = 6
      IF(KI.NE.1)GO TO 350
      WRITE(6,302)K,NN,N,IUP,XZ(N, 1), XZ(N, 2), XZ
         (N,3),
```

```
     1A,AK1,AK2,GA,AN,V,V1
302  FORMAT(4I5,3F10.3,4F10.4,3F10.2)
350  IF(ABS(XZ(N,1)).LE.ABS(XMAXR).AND.ABS(XZ
       (N, 1)).GE.ABS(XMINR).AND.
       ABS(XZ(N,2)).LE.ABS(YMAXR).AND.ABS (XZ
       (N,1)).GE.ABS(YMINR).AND.N.LT.100)
     2GO  TO  354
     IF(IUP.EQ.1)OUT1=10.0
     IF(IUP.NE.1)OUT2=10.0
     IF(KI.NE.1)  GO TO 407
     EN=E(N)·108/3.141592654
     WRITE(6,351)NN,IUP,N,XZ(N,1),XZ(N,2),XZ(N,3)
351  FORMAT(2X,2HNN,I3,5X,3HIUP,I2,5X,1HN,I3,5X,
       3HXZ1,F7.3,5X,3HXZ2,F7.13,5X,3HXZ3,F7.3,5X,
       1HE,F6.2)
     GO TO 407
354  IF(A0(K+1,2).NE.0.1)  GO TO 357
     VX=AC(K+1,4)
     IF(IUP.EQ.1)  GO TO 355
     VZ=AC(K+1,5)
     GO TO 3514
355  VZ=-AC(K+1,5)
     GO TO 3514
357  IF(A0(K+1,2).NE.1.0)  GO TO 359
     DV=0.0
     E(N)=AN
     A=0.0
     V=AC(K+1,1)
     SC=R*COS(E(N))
     SD=R*SIN(E(N))
     IF(IUP.NE.1)  GO TO 358
     XZ(N+1,1)=XZ(N,1)+SC
```

```
      XZ(N+1,2)=XZ(N,2)-SD
      GO TO 3502
 358  XZ(N+1,1)=XZ(N,1)+SC
      XZ(N+1,2)XZ(N,2)+SD
3502  DT=R/AC(K+1,1)
      XZ(N+1,3)=XZ(N,3)+DT
      GO TO 3506
 359  ZN12=Z(XZ(N,1),K+1)
      ZN11=Z(XZ(N,1),K)
      DZ=ZN12-ZN11
      DV=AC(K+1,2)-AC(K+1,1)
      IF(IUP.EQ.1)  VZ=-DV/DZ
      IF(IUP.NE.1)  VZ=DV/DZ
      DX=1.0
      X1=XZ(N,1)+DX
      ZX11=Z(X1,K)
      ZX12=Z(X1,K+1)
      DZ1=ZX12-ZX11)
      XN11=Z(XZ(N,1),K)
      DVX=DV*((XZ(N,2)-ZX11)/DZ1-(XZ(N,2)-XN11
        /DZ)
      VX=DVX/(DX+0.000001)
3514  GA=ABS(SQRT(VX*VX+VZ*VZ))
      AK2=VZ/(VX+0.000001)
      IF(IUP.NE.1)  GO TO 363
      E(N)=AN+N*Q
      AK1=-TAN(E(N))
      GO TO 3503
 363  E(N)=AN-N*Q
      AK1=TAN(E(N))
3503  A=ATAN((AK2—AK1)/(1+AK1*AK2))
      A=ABS(A)
```

```
    9 FORMAT(12F10.5)
      ZN13=Z(XZ(N,1),K)
      IF(A0(K+1,2).NE.0.1)  GO TO 365
      V=AC(K+1,1)+SQRT((AC(K+1,4)*XZ(N,2))**
         2+(AC(K+1,5)*(XZ(N,2)-ZN13))**2)
      GO TO 367
  365 V=AC(K+1,1)+DV*(XZ(N,2)-ZN13)/DZ
  367 IF(ABS(A).GE.0.001)  GO TO 379
      E(N)=ATAN(AK2)
      SC=R*COS(E(N))
      SD=R*SIN(E(N))
      IF(IUP.NE.1)  GO TO 373
      XZ(N+1,1)=XZ(N,1)+SC
      XZ(N+1,2)=XZ(N,2)-SD
      ZN21=Z(XZ(N+1,1),K)
      IF(A0(K+1,2).EQ.0.1)GO TO 369
      ZN22=Z(XZ(N+1,1),K+1)
      DZ1=ZN22—ZN11
      V1=AC(K+1,1)+DV*(XZ(N+1,2)—ZN21)/DZ1
      GO TO 371
  369 V1=AC(K+1,1)+SQRT((AC(K,4)*XZ(N+1,2))**2
         +(AC(K,5)*(XZ(N+1,2)-ZN21)1)**2)
  371 DT=ALOG(ABS(V/V1))/(GA+0.000000001)
      GO TO 3505
  373 XZ(N+1,1)=XZ(N,1)+SC
      XZ(N+1,2)=XZ(N,2)+SD
      ZN21=Z(XZ(N+1,1),K)
      IF(A0(K+1,2).EQ.0.1)  GO TO 375
      ZN22=Z(XZ(N+1,1), K+1)
      DZ1=ZN22-ZN11
      V1=AC(K+1,1)+DV*(XZ(N+1,2)-ZN21)/DZ1
      GO TO 377
```

```
375  V1=AC(K+1,1)+SQRT((AC(K+1,4)*XZ(N+1,1))**
         2+)AC(K+1,5)*XZ(N+1,2)-Z1N21))**2)
377  DT=ALOG(ABS(V1/V))/(GA+0.000000001)
3505 XZ(N+1,3)=XZ(N,3)+DT
     GO TO 3506
379  R0=V/(GA*SIN(A)+0.000000001)
     R0=ABS(R0)
     CALL XY(Q)
     E(N+1)=AN-(N+1)*Q
     IF(IUP.NE.0.OR.ABS(E(N+1)).GT.0.01745) GO TO
         3506
     DO 381 I0=1,3
381  XZO(NN,I0)=XZ(N+1,I0)
     GO TO 407
3506 IF(IUP.NE.1)  GO TO 385
     H0=Z(XZ(N+1,1),K)
     H1=Z(XZ(N+1,1),K+1)
     IF(XZ(N+1,2).GE.H1)  GO TO 3511
     IF(XZ(N+1,2).LT.H0)  GO TO 3507
     N=N+1
     GO TO 3501
385   H0=Z(XZ(N+1,1),K+1)
     IF(XZ(N+1,2).GT. H0)GO TO 3507
     IF(KI.NE.1)  GO TO 5
     WRITE(6,  3  )  H0,XZ(N+1,1),XZ(N+1,2)
   3 FORMAT(//,5X,2HH0,F10.3,5X,3HXZ1, F10.3,5X, 3H
     XZ2,F10.3)
   5 N=N+1
     GO TO 3501
3507 IF(ABS(XZ(N+1,2)-H0).GE.0.01)  GO TO 392
     IF(IUP.NE.1)GO TO 389
     DO 387 I0=1,3
```

```
 387  XZK(K,I0)=XZ(N+1,I0)
      GO TO 3510
 339  DO 390 I0=1,3
 390  XZO(NN,I0)=XZ(N+1,I0)
      GO TO 3510
 392  IF(DV.NE.0.0 • OR.A.NE.00)  GO,TO 391
      F1=0.0
      F2=R
      IF(RI.EQ.1)WRITE(6,301)XZ(N+1,1),XZ(N+1.2),H
         0,K
 301  FORMAT(12F10.3)
      GO TO 3509
 391  F1=0.0
      F2=Q
      IF(KI.EQ.1) WRITE(6,301) XZ(N+1,1), XZ(N+1,2)
         H0,K
3509  CALL SEARC(F1,F2)
      IF(DV.NE.0.0.OR  .A.NE.0.0)  GO TO 399
      E(N+1)=AN
      IF(IUP.NE.1)  GO TO 395
      XZ(N+1, 1)=XZ(N,1)+SS*COS(E(N))
      XZ(N+1,2)=Z(XZ(N+1,1),K)
      XZ(N+1,3)=XZ(N,3)+SS/(V+0.000000001)
      DO 393 I0=1, 3
 393  XZK(K,I0)=XZ(N+1,I0)
      GO TO 3510
 395  XZ(N+1,1)=XZ(N,1)+SS*COS(E(N))
      XZ(N+1,2)=Z(XZ(N+1, 1),K+1)
      XZ(N+1,3)=XZ(N,3)+SS/(V+0.000000001)
      DO 397 I0=1,3
 397  XZO(NN,I0)=XZ(N+1,I0)
      GO TO 3510
```

```
399  CALL XY(SS)
     IF(IUP.NE.1)  GO TO 403
     E(N+1)=E(N)+SS
     DO 401 I0=1,3
401  XZK(K,I0)=XZ(N+1,I0)
     GO TO 3510
403  DO 405 I0=1,3
405  XZO(NN,I0)=XZ(N+1,I0)
     E(N+1)=E(N)-SS
3510 CALL ANGLE(K)
407  IF(IFL. EQ.10)  GO TO 3511
     IF(KFR.EQ.1)  CALL FIGR
3511 CONTINUE
     RETURN
     END
C
C       THE UPWARD SEISMIC RAYS ABOVE THE AIM
C       LAYER IS TRACED
     SUBROUTINE UPTXZ(N1,NN1, TX, TX, TX1)
     DIMENSION TX(200,2,),TX1(200,2)
     COMMON /NR1/ NR,NNR.IP,LS
     COMMON /K1/ K
     COMMON /EN  /LG,LF,LR
     COMMON /L1/ LL
     COMMON /ID/ KSP,IFL
     COMMON /OUT/ OUT1,OUT2
     COMMON /TN/ TAPE,NNS
     COMMON /ZAN/ MTAN,MM,SH1,SH2,N,NN,
        IUP,IO,KI,KZ,MM1,MAC
     COMMON /ZAN1/ AN,AO,Q,DT,DV,GA,A,R0,R,SS
     COMMON /ANG/ B11(500),B12(500),B21(50)
     COMMON /XZT/ XZ(500,3),XZO(500,3)XZK(50,3),
```

```
     E(500)
   N2=N1
   IF(K.LE.1)  GO TO 4502
   KUP=K
   DO 455 J=2,KUP
   II=2+KUP-J
   DO 451 I0=1,500
   DO 451 J0=1,3
451 XZ(I0,J0)=0.0
   AN=B21(II)
   N=1
   DO 453 I0=1,3
453 XZ(1, I0)=XZK(II, I0)
   K=II-1
   IUP=1
   CALL TRACE(K)
   IF(OUT1.EQ.10.0.AND.LL.EQ.1)  GO TO 4501
   IF(OUT1.NE.10.0)  GO TO 454
   IF(AO.NE.0.0)  NR=NR-1
   GO TO 4501
454 TX(NN1,1)=XZK(II-1,1)
   TX(NN1,2)=XZK(II-1,3)
   CALL MOVE(NN1,TX,TX1)
455 CONTINUE
4501 K=KUP
4502 CONTINUE
   RETURN
   END
C
C      THE UPWARD SEISMIC RAYS IN THE AIM LA-
C      YER IS TRACED
   SUBROUTINE UPTXO(N1,NN1,TX,TX1)
   DIMENSION TX(200,2),  TX1(200,2)
```

```
      COMMON  /NR1/  NR  NNR,IP,LS
      COMMON  /K1/  K
      COMMON  /TN/  TAPE,NNS
      COMMON  /OUT/  OUT1,OUT2
      COMMON  /ZAN/  MTAN,MM,SH1,SH2,N,NN,
         IUP,IO,KI,KZ,MM1,MAC
      COMMON /ZAN1/ AN,AO,Q,DT,DV,GA,R0,R,SS
      COMMON  /XZT/  XZ(500,3),XZO(500,3),XZK(50,3),
         E(500)
      DO 461 I0=1,3
 461  XZ(1,I0)=XZO(N1,I0)
      IUP=1
      CALL TRACE(K)
      IF(OUT1.NE.10.0)  GO TO 462
      IF(AO.NE.0.0)  NR=NR-1
      GO TO 4601
 462  DO 463 I0=1,3
 463  XZK(K,I0)=XZ(N+1,I0)
      TX(NN1,1)=XZK(K,1)
      TX(NN1,2)=XZK(K,3)
      CALL MOVE(NN1,TX,TX1)
4601  CONTINUE
      RETURN
      END
C
C        THE SUBPROGRAM OF INTERPOLATION MET-
           HOD
      SUBROUTINE ZAH(N,X,T,U,TF)
      DIMENSION X(N),T(N)
      NM1=N-1
      DO 10 I=2,NM1
      IF(U.LE.X(I))  GO TO 20
```

```
 10  CONTINUE
     I=NM1
     GO TO 30
 20  IF(I.EQ.2)  GOTO 30
     IF(U-X(I-1).LT.X(I)-U)  I=I-1
 30  X1=X(I-1)
     X2=X(I)
     X3=X(I+1)
     A1=(U-X2)*(U-X3)/((X1-X2)*(X1-X3))
     A2=(U-X3)*(U-X1)/((X2-X3)*(X2-X1))
     A3=(U-X1)*(U-X2)/((X3-X1)*(X3-X2))
     TF=A1*T(I-1)+A2*T(I)+A3*T(I+1)
     RETURN
     END
C    COMPUTING THE THEORETICAL TRAVEL—TIME
     SUBROUTINE AQ(N1,FT,PX,IM,TXX,TXT)
     DIMENSION FT(IM),PX(IM)
     DIMENSION TXX(N1),TXT(N1)
     COMMON /ZAN/ MTAN,MM,SH1,SH2,N,NN,IUP,
        IO,KI,KZ,MM1,MAC
     COMMON  /EN/  LG,LF,LR
     COMMON/FXF/TX(200,2),TX1(200,2),ALL(200,2),
        ALL1(200,2),
     TXR(200,2),TXR1(200,2)
     DO 483 I=1,N1
481  IF(LF.EQ.1)TXX(I)=TX(I,1)
     IF(LF.EQ.1)TXT(I)=TX(I,2)
     IF(LG.EQ.1)TXT(I)=ALL(I,2)
     IF(LG.EQ.1)TXX(I)=ALL(I,1)
483  CONTINUE
     DO 485I=1, IM
     VQ=0.0
```

```
      CALL ZAH(N1,TXX,TXT,PX(I),VQ)
485   FT(I)=VQ
      RETURN
      END
C
C     CONTROLLING THE WHOLE SEISMIC RAYS TRA-
C     CE PROCESS
      SUBROUTINE ZYFT(FT,PX,IM)
      DIMENSION FT(IM), PX(IM) ,XZL(3)
      COMMON  /NR1/  NR,NNR,IP,LS
      COMMON  /K1/  K
      COMMON  /L1/  LL
      COMMON  /OUT/  OUT1,OUT2
      COMMON  /ANG1/  ALP
      COMMON  /TN/  TAPE,NNS
      COMMON  /ID/  KSP,IFL
      COMMON  /FAN/  KK,K0,KO,LN
      COMMON  /EN/  LG,LF,LR
      COMMON  /FAN1/  M,MZG,MZF,IC,IB,MAX,FO,F
      COMMON  /KF/  KFR,KFT,KFZ
      COMMON  /FIG/  GLX,LX,GLY,LY,XMINR,
       XMAXR,YMINR,YMAXR
      COMMON  /FIG1/  GLXT,LXT,GLYT,LYT,XMINT,
        XMAXT,YMINT,YMAXT
      COMMON  /ZAN/  MTAN,MM,SH1,SH2,N,NN,
        IUP,IO,KI,KZ,MM1,MAC
      COMMON /ZAN1/ AN,AO,Q,DT,DV,GA,A,R0,R,SS
      COMMON  /XZT/  XZ(500,3),XZO(500,3),XZK,
        (50,3),E(500)
      COMMON  /ANG/  B11(500),B12(500),B12(50)
      COMMON  /ZAC/  AC1(10,50),AC(10,50),A0(10, 50)'
        IX(11,51)
```

```
      COMMON/FXF/TX(200,2),TX1(200, 2), ALL(200, 2),
         ALL1(200,2),
      TXR(200,2),TXR1(200,2)
      DIMENSION TXX(100),TXT(100)
      K=1
      IF(KZ.EQ.0) K=KO
      NNK=NNS
      Q=0.01745
      DL=5.0
      IFL=1
      DO 501 I0=1,500
      DO 501 J0=1,3
 501  XZO(I0,J0)=0.0
5001  NN=0
      NR=0
      LL=0
      NNR=0
      NNS=NNK
      IL=30
      I1=0
      IF(KFZ.EQ.1) GO TO 502
      FX=GLX-1.0
      FY=GLYT+GLY*2+12.0
      IF(KFT.EQ.5)FY=GLY+7.0
      IF(SH2.GE.0.0)CALL PLOT(1.0,FY,-3)
      IF(SH2.LT.0.0)CALL PLOT(-FX,FY,-3)
 502  DO 503 I0=1,500
      DO 503 J0=1,3
 503  XZ(I0,J0)=0.0
      DO 505 I0=1,50
      DO 505 J0=1,3
 505  XZK(I0,J0)=0.0
```

```
      DO 507 I0=1,200
      DO 507 J0=1,2
      TX(I0,J0)=0.0
      TX1(I0,J0)=0.0
      TXR(I0,J0)=0.0
      TXR1=(I0,J0)=0.0
      ALL(I0,J0)=0.0
 507  ALL1(I0,J0)=0.0
5002  NN=NN+1
      NR=NR+1
      IF(NN.LE.NNS)  GO TO 508
      IF(TXR(1,1).NE.0.0)  NNR=NNR-1
      IF(LF.EQ.1)  NR=NR-1
      GO TO 5010
 508  DO 509 J0=1,3
 509  XZ(1,J0)=XZO(NN,J0)
      KKP=K
      PKK=KKP
      DO 555 K=1, KKP
      WK=0.0
      IF(K.NE.KKP)  WK=10.0
      IF(K.NE.1)  GO TO 552
      DO 551 I0=1,200
      DO 551 J0=1,3
 551  XZ(I0,J0)=0.0
      GO TO 554
 552  DO 553 J0=1,3
 553  XZ(1,J0)=XZO(NN,J0)
 554  AO=0.0
      IUP=0
      N=1
      IF(K.EQ.1)AN=ATAN(AC(KKP+1,6)*PKK/(TAPE*
```

```
       FLOAT(NN)))
     IF(K.EQ.1.AND.A0(K+1,2).NE.1.0.AND.KKP.
       EQ.1)
       AN=1.57-FLOAT(NN)*1.57/(FLOAT(NNK)+2.0)
     IF(K.NE.1)  AN=B12(NN)
     CALL TRACE(K)
     IF(OUT2.NE.10.0)  GO TO 555
     NR=NR-1
     GO TO 514
555  CONTINUE
514  K=KKP
     IF(OUT2.EQ.10.0)  GO TO 5002
     IF(E(N+1).LT.0.01745.AND.AO.EQ.0.0)  GO TO
       5005
     IF(AO.LT.0.0)  GO TO 5003
     IF(AO.GT.0.0)  I1=I1+1
     IF(I1.EQ.1)  NL=NN
     IF(I1.GT.1)  GO TO 5004
     IF(LG.NE.1  )  GO TO 5004
     B1=B11(NN)
     DO 513 I0=1,3
513  XZL(I0)=XZO(NN,I0)
     DO 519 I=1,100
     IF(K.NE.1)  GO TO 520
     AN=AN+0.04
     ANL=AN
     IUP=0
     N=1
     DO 515 I0=1,500
     DO  515 J0=1,3
515  XZ(I0,J0)=0.0
     DO 517 I0=1,3
```

```
517  XZK(NN,I0)=0.0
     IFL=10
     CALL TRACE(K)
     IF(OUT2.EQ.10.0)  GO TO 5002
     IF(AO.LT.0.0)  GO TO 520
519  CONTINUE
520  TL=0.0
     IFL=1
     IF(KFT.EQ.5)  GO TO 522
     IF(KFR.NE.1)  GO TO 522
     CALL PLOT(0.0,  0.0,  3)
     FD=GLY+GLYT+10.0
     CALL PLOT(0.0, -FD, -3)
522  KKP=K
     DO 561 K=1,KKP
     IF(K.NE.1)  GO TO 563
     DO 565 I0=1,200
     DO 565 J0=1,3
565  XZ(I0,J0)=0.0
     GO TO 567
563  DO 569 J0=1,3
569  XZ(1,J0)=XZO(NN,J0)
567  IUP=0
     IF(K.EQ.1.AND.KKP.NE.1.AND.A0(K, 2).EQ.1)
        AN=ATAN(AC(KKP+1,6)*PKK/(TAPE*FLOAT
        (NN)))
     IF(K.EQ.1.AND.A0(K+1,2).NE.1.0.AND.KKP.E
        Q.1)
       AN=1.57-FLOAT(NN)*1.57/(FLOAT(NNK)+2.0)
     IF(K.NE.1)AN=B12(NN)
     IF(KKP.EQ.1)  AN=ANL
     N=1
```

```
      CALL TRACE(K)
      IF(OUT2.EQ.10.0)  GO TO 560
561   CONTINUE
560   K=KKP
      IF(OUT2.EQ.10.0)  GO TO 5002
      IP=1
      LS=1
666   ALL(IP,1)=XZO(NN,1)+IP*DL
      LL=1
      ALL(IP+1,1)=ALL(IP,1)+DL
      ZL21=Z(ALL(IP+1,1),K+1)
      ZL12=Z(ALL(IP,1),K+1)
      HH=ZL11-ZL12
      GAM=ATAN(HH/DL)
      IF(IP.EQ.1)TL=0.0
      IF(IP.NE.1) TL=TL+SQRT(HH*HH+DL*DL)/AC
        (K+1,3)
      IF(IP.EQ.1)  ALL(IP,2)=XZO(NN,3)
      IF(IP.NE.1)  ALL(IP,2)=TL+XZO(NN,3)
      DO 521 I0=1,500
      DO 521 J0 1,3
521   XZ(I0,J0)=0.0
      N=1
      AN=1.57079634-(ALP+GAM)
      XZ(1,1)=ALL(IP,1)
      XZ(1,2)=ZL12
      XZ(1,3)=ALL(IP,2)
      IUP=1
      CALL TRACE(K)
      IF(OUT1.NE.10.0)  GO TO 526
      LS=LS-1
      GO TO 525
```

```
526  DO 523 I0 = 1, 3
523  XZK(K,I0) = XZ(N + 1, I0)
     ALL(IP, 1) = XZ(N + 1, 1)
     ALL(IP, 2) = XZ(N + 1, 3)
     CALL MOVE(IP, ALL, ALL1)
     CALL UPTXZ(IP, IP, ALL, ALL1)
     IF(OUT1.NE.10.0)  GO TO 525
     LS = LS - 1
525  IP = IP + 1
     LS = LS + 1
     IF(IP.LT.IL)  GO TO 666
     LS = LS - 1
     LL = 0
     IF(KFT.EQ.5)  GO TO 528
524  IF(KFR.NE.1)  GO TO 528
     CALL PLOT(0.0, - GLY, - 3)
     CALL PLOT(0.,0.,3)
     CALL RAH(GLX, LXT, GLY, LY, 0.2, 2, XMINR,  XMA
     XR, YMINR, YMAXR)
     K2 = K + 1
     CALL FIGZ  (K2)
     CALL PLOT(0.0,   0.0  , 3)
     FYT = GLY + 5.0
     CALL PLOT(0.0,   FYT, - 3)
     CALL RAH(GLXT, LXT, GLYT, LYT, 0.2, 1, XMINT,
        XMAXT, YMINT, YMAXT)
     IF( IO.EQ.0.AND.LS.GE.2)     CALL FIGT(ALL,
        LS, K)
     IF( IO.NE.0.AND.LS.GE.2)     CALL FIGT(A LL1,
        S, K)
     CALL PLOT(0.0, 0.0, 3)
     FY = GLYT + GLY + 5.0
```

```
      CALL PLOT (0.0,    FY,   -3)
 528  NL=NN
      B11(NN)=B1
      DO 527  I0=1,3
 527  XZO(NN,I0)=XZL(I0)
      GO TO 5004
5003  DO 592 I0=1,500
      DO 592 J0=1,3
 592  XZ(I0,J0)=0.0
      N=1
      AN=B11(NN)
      CALL UPTXO(NN,NR,TX,TX1)
      IF(OUT1.EQ.10.0)  GO TO 530
      CALL UPTXZ(NN,NR,TX,TX1)
 530  NL=NN
      IF(NN.LT.NNS)  GO TO 5002
      GO TO 5010
5004  IF(LF.NE.1)  GO TO 5010
      IF(NL.GE.NNK)  GO TO 5010
      IF(WK.EQ.10.0)  GO TO  5002
      DO 531 I0=1,500
      DO 531  J0=1,3
 531  XZ(I0,J0)=0.0
      N=1
      AN=B11(NN)
      CALL UPTXO(NN,NP,TX,TX1)
      IF(OUT1.EQ.10.0)  GO TO 532
      CALL UPTXZ(NN,NR,TX,TX1)
 532  IF(NN.GE.NNS)  GO TO 5010
      GO TO 5002
5005  DO 533 I0=1,500
      DO 533 J0=1,3
```

```
533  XZ(I0,J0)=0.0
     AN=0.0
     N=1
     NR=NR-1
     NNR=NNR+1
     CALL UPTXO(NN,NNR,TXR,TXR1)
     IF(OUT1.NE.10.0)  GO TO 534
     NNR=NNR-1
     GO TO 536
534  CALL UPTXZ(NN,NNR,TXR,TXR1)
     IF(OUT1.NE.10.0)  GO TO 536
     NNR=NNR-1
536  IF(NN.LT.NNS)  GO TO 5002
5010 IF(KFT.NE.5)  GO TO 538
     IF(IO.EQ.0.AND.LS.GE.3) CALL FIGT(ALL,LS,K)
     IF(IO.NE.0.AND.LS.GE.3) CALL FIGT(ALL1,
        LS,K)
     IF(IO.EQ.0.AND.NNR.GE.3)  CALL FIGT(TXR,
       NNR,K)
     IF(IO.NE.0.AND.NNR.GE.3)  CALL FIGT(TXR1,
       NNR,K)
     IF(IO.NE.0.AND.NR.GE.3)  CALL FIGT(TX1, N-
        R,K)
     GO TO 540
538  IF(KZ.EQ.0)  GO TO 545
540  WRITE(6,535)  K
535  FORMAT(//,2H* *,I2,2X,5HLAYER,2H* *)
     WRITE(6,537)NL
537  FORMAT(//,5X,6HLITTLE,1X,5HANGLE,1X,15HREF
        ECTIVE-WAVE,1X,2HNL,I3)
     WRITE(6,539)
539  FORMAT(//,5X,15HREFLECTIVE-WAVE)
```

```
      WRITE(6,702)
702   FORMAT(/,12X,1HT,9X,1HT,9X,1HX,9X,1HT, 9X,1
          HX,9X,IHT,9,
          1HX,9X,IHT,9X,1HX,9X,1HT,9X,1HX,9X,1HT/)
541   FORMAT(5X,12F10.3)
      IF(IO.EQ.0)  WRITE(6,541)  ((TX  (I,J),J=1,2),
        I=1,NR)
      IF(IO.NE.0)  WRITE(6,541)  ((TX1  (I,J),J=1,2),
        I=1,NR)
      WRITE(6,543)
543   FORMAT(//,5X,15HREFRACTIVE-WAVE)
      WRITE(6,702)
      IF(IO.EQ.0)  WRITE (6,541)  ((  ALL(I,J),J=1,2 ),
        I=1,LS)
      IF(IO.NE.0)  WRITE  (6,541)  ((ALL1(I,J),J=1,2),
        I=1,LS)
      WRITE(6,544)
544   FORMAT(//,5X,11HDIVING-WAVE)
      WRITE(6,702)
      IF(IO.EQ.0.AND.TXR(I,1).NE.0.0)  WRITE(6,5
          ((TXR(I,J),J=1,2), I=1,NNR)
      IF(IO.NE.0.AND.TXR1(1,1).NE.0.0)  WRITE(6,541)
          ((TXR1(I,J),J=1,2),I=1,NNR)
      IF(KFR.NE.1)  GO TO 548
      CALL  PLOT(0.0,0.0,3)
      CALL  PLOT(0.0,-GLY,-3)
4     FORMAT(//,5X,3HGLX,F8.3,2X,2HLX,I2,2X,3HGLY,
          F8.3,2X,2HLY,I2,2X
      15HXMINR,F8.3,2X,5HXMAXR,F8.3,2X,5HYMINR,
          F8.3,2X,5HYMAXR,F8.3)
      CALL  RAH(GLX,LXT,GLY,LY,0.2,2,XMINR,XMAX
          R,YMINR,YMAXR)
```

```
      K2=K+1
      CALL FIGZ  (K2)
  548 IF(KFT.NE.1) GO TO 545
      FY=GLY+5.0
      CALL PLOT  (0.0,    FY,   -3)
      CALL RAH(GLXT,LXT,GLYT,LYT,0.2,1,XMINT,X-
         MAXT,YMINT,YMAXT)
      IF(IO.EQ.0.AND.NNR.GE.2)  CALL FIGTR (TXR,
         NNR,K)
      IF(IO.NE.0.AND.NNR.GE.2)  CALL FIGTR(TXR1,
         NNR,K)
      IF(IO.EO.0.AND.NR'GE.2) CALL FIGT (TX,NR, K)
      IF(IO.NE.0.AND.NR.GE.2)              CALL FIGT
        (TX1,NR,K)
      CALL PLOT  (0.0,0.0,3)
      FX=GLX+10.0
      FY=GLY * 2+GLYT+15.2
      CALL PLOT(FX, -FY, -3)
  545 K=K+1
      IF(K.GT.KO) GO TO 5011
      GO TO 5001
 5011 IF(KZ.NE.0) GO TO 5012
      IF(KZ.NE.0.) GO TO 5012
      IF(LF.EQ.1)  CALL AQ(NR,FT,PX,IM,TXX,TXT)
      IF(LG.EQ.1)  CALL AQ(LS,FT,PX,IM,TXX,TXT)
 5012 CONTINUE
      RETURN
      END
C
C     TO ESTABLISH JACOBI MATRIX
      SUBROUTINE BCAJ(CAJ(CAJ,FT,FTO,PX,NNX,
```

```
      IM)
      DIMENSION  CAJ(IM,NNX),FT,(IM),FTO(IM),PX
     (IM)
      COMMON /FY/  AY1(5,21),AY2(5,21),DDO(21),XDJ
     (21)
      COMMON  /FAN1/  M,MZG,MZF,IC,IB,MAX
      COMMON  /FAN/  KK,K0,KO,LN
      COMMON  /ZAN/  MTAN,MM,SH1SH2,N,NN,IUP,
     IO,KI,KZ,MM1,MAC
      COMMON/ZAC/AC1(10,50),AC(10,50),A0(10,50),IX
     (11,51)
      IB=0.0
      DO 551 IA=2,MAX
      IF(IX(K0,IA).NE.1)  GO TO 551
      AC(K0,IA-1)=AC(K0,IA-1)+XDJ(IA-1)
      CALL ZYFT(FT,PX,IM)
      IB=IB+1
      DO 550 J=1,IM
550   CAJ(J,IB)=(FT(J)-FTO(J))/XDJ(IA-1)
      AC(K0,IA-1)=AC(K0,IA-1)-XDJ(IA-1)
551   CONTINUE
      RETURN
      END
C
C          COMPUTING RESIDUAL AND SUM OF SQUARES
      SUBROUTINE QF(FT,F,DPT,PT,IM)
      DIMENSION FT(IM),DPT(IM),PT(IM)
      COMMON  /SE/  S
      S1=0.0
      F=0
      DO 561 I=1,IM
```

```
      DPT(I)=PT(I)-FT(I)
      S1=S1+ABS(DPT(I))
  561 F=F+DPT(I)*DPT(I)
      S=S1/IM
      S=ABS(S)
      RETURN
      END
C
C     PREVENTED THE INTERFACES FROM CROSSING
      SUBROUTINE OUTZ
      COMMON  /FAN/  KK,K0,KO,LN
      COMMON /ZAN/  MTAN,MM,SH1,SH2,N,NN,IUP,
         IO,KI,KZ,MM1,MAC
      COMMON  /FIG/  GLX,LX,GLY,LY,XMINR,XMA
         XR,YMINR,YMAXR
      COMMON  /ZAC/  AC1(10,50),  AC(10,50),  A0(10,
         50),IX(11,51)
      IF(K0.GE.MM1)  GO TO 502
      FX=2.0
      JX=(XMAXR-XMINR)/FX
      JX=JX+2
      B1=0.0
      C1=0.0
      DO 501 I=1,JX
      XX=FX*FLOAT(I)
      ZA=Z(XX,KO)
      ZB=Z(XX,K0)
      ZC=Z(XX,K0+1)
      BB=ZB-ZA
      IF(BB.LT.B1) B1=BB
      CC=ZC-ZB
```

```
      IF(CC.LT.C1)  C1=CC
  501 CONTINUE
      IF(B1.LT.0.0)  AC(K0,6)=AC(K0,6)-B1+0.5
      IF(C1.LT.0.0)  AC(K0,6)=AC(K0,6)+C1-1.0
  502 CONTINUE
      RETURN
      END
C     DAMPING LEAST SQUARES WITH CONSTRAINED
      SUBROUTINE MAQT(FT,FTO,PX,PT,DPT,IM,DDP,
        R1,XO,X,AW,NX,AZE,NNX,CAJ)
      DIMENSION FT(IM),  FTO(IM),  PX(IM),  PT
        (IM),  DPT(IM),DDP(NX),
        R1( NX),  XO(NX),  AZE(NNX,NNX),  CAJ
        (  IM,NNX),AW(NX),X(NX)
      DIMENSION IS(50)
      COMMON  /SE/  S
      COMMON  /KF/  KFR,KFT,KFZ
      COMMON  /NR1/  NR,NNR,IP,LS
      COMMON  /FIG/  GLX,LX,GLY,LY,XMINR,XMA-
        XR,YMINR,YMAXR
      COMMON  /FIG1/  GLXT,LXT,GLYT,LYT,XMINT,
        XMAXT,YMINT,YMAXT
      COMMON/ID/KSP,IFL
      COMMON  /IHFG/  IHG,IHF,ITG,ITF
      COMMON  /EN/  LG,LF,LR
      COMMON  /KZ1/  KC,KX,KY,KD,KJ
      COMMON /ZAN1/ AN,AO,Q,DT,DV,GA,A,R0,R,SS
      COMMON  /FAN/  KK,K0,KO,LN
      COMMON  /FAN1/  M,MZG,MZF,IC,IB,MAX
      COMMON/ZAN/MTAN,MM,SH1,SH2,N,NN,IUP,
        IO,KI,KZ,MM1,MAC
      COMMON/ZAC/AC1(10,50),AC(10,50),A0(10,50), IX
```

```
     (11,51)
   COMMON /ST/ IH(500),PGJ(100),GST(500),FST(500)
   COMMON  /XZT/  XZ(500,3),XZO(500,3),XZK(50,3),
     E(500)
   COMMON  /ANG/  B11(500),B12(500),B21(50)
   COMMON  /FXF/  TX(200,2),TX1(200,2),ALL(200,2),
     ALL1(200,2),
     TXR(200,2),TXR1(200,2)
   COMMON /FY/  AY1(5,21),AY2(5,21),DDO(21),XDJ
     (21)
   FO=0.0
   F=0.0
   DO 601 I0=1,NX
601 XO(I0)=0.0
   DO 603 I0=1,MAC
603 IS(I0)=0.0
   IDED=0
   D=1.0
   I=1
   DO 605 J =1,MAC
   IF(IX(K0,J+1).NE.1)  GO TO 605
   XO(I)=AC(K0,J)
   I=I+1
605 CONTINUE
   DO 607 I=1,IM
   IF(LG.NE.1)  GO TO 11
   IHG=IHG+1
   IE=IH(IHG)
   PX(I)=PGJ(IE)
   ITG=ITG+1
   PT(I)=GST(ITG)
   GO TO 607
```

```
11   IF(LF.NE.1)  GO TO 607
     IHF = IHF + 1
     IE = IH(IHF)
     PX(I) = PGJ(IE)
     ITF = ITF + 1
     PT(I) = FST(ITF)
607  CONTINUE
     I = 1
     DO 614 J = 1, MAC
     IF(IX(K0, J + 1).NE.1.0)  GO TO 614
     DDP(I) = DDO(I) * 0.5
     I = I + 1
614  CONTINUE
     IF(KI.NE.5)  GO TO 611
     WRITE(6, 608)
608  FORMAT(//, 5X, 3HCHE, 1X, 4HLIAN)
702  FORMAT(/, 12X, 1HX, 9X, 1HT, 9X, 1HX, 9X, 1HT, 9X,
        1HX, 9X, 1HT, 9X,
        1HX, 9X, 1HT, 9X, 1HX, 9X, 1HT, 9X, 1HX, 9X, 1HT/)
     WRITE(6, 702)
     WRITE(6, 609) (PX(I), PT(I), I = 1, IM)
609  FORMAT(5X, 12F10.3)
     WRITE(6, 610)
610  FORMAT(//, 5X, 2HAC)
     WRITE(6, 609) (AC(K0, I), I = 1, MAC)
     CALL OUTZ
611  CALL ZYFT(FTO, PX, IM)
     IF(KI.NE.5)GO TO 612
     WRITE(6, 702)
541  FORMAT(5X, 12F10.3)
     WRITE(6, 541)  ((TX(I, J), J = 1, 2), I = 1, NR)
612  CALL QF(FTO, FO, DPT, PT, IM)
```

```
      IF(KZ.EQ.5)  GO TO 785
6002  DO 613 I0=1,NNX
      DO 613 J0=1,NNX
 613  AZE(I0,J0)=0.0
      IDED=IDED+1
      DO 616 J=1,MAC
 616  DDP(I)=DDP(I)*0.5
      CALL BCAJ(CAJ,FT,FTO,PX,NNX,IM)
      DO 615 I=1,IM
 615  CAJ(I,NNX)=DPT(I)
      DO 619 I=1,NX
      DO 617 J=1,NNX
      DO 617 KK=1,IM
 617  AZE(I,J)=AZE(I,J)+CAJ(KK,I)*CAJ(KK,J)
 619  R1(I)=SQRT(AZE(I,I))
      IF(KI.NE.5)  GO TO 634
      WRITE(6,621)  IDED
 621  FORMAT(//,5X,5HIDED,I3)
      WRITE(6,623)
 623  FORMAT(//,5X,3HFTO )
      WRITE(6,625)  (FTO(I),I=1,IM)
 625  FORMAT(12F10.3)
      WRITE(6,627)
 627  FORMAT(//,5X,2HXO )
      WRITE(6,625)(XO(I),I=1,NX)
      WRITE(6,629)
 629  FORMAT(//,5X,3HCAJ  )
      DO 630 I=1,IM
 630  WRITE(6,631)    (CAJ(I,J),J=1,NNX)
 631  FORMAT(12E10.3)
      WRITE(6,633)
 633  FORMAT(//,5X,3HAZE  )
```

```
      DO 632 I=1,NNX
632   WRITE(6,631)    (AZE(I,J),J=1,NNX)
634   DO 637 I=1,NX
      I1=I+1
      IF(NX.LT.I1) GO TO 111
      DO 635 J=I1,NX
635   AZE(I,J)=AZE(I,J)/(R1(I)*R1(J))
111   AZE(I,NNX)=AZE(I,NNX)/R1(I)
637   CONTINUE
6003  I0=1
      DO 643 J=1,MAC
      IF(IX(K0,J+1).NE.1)  GO TO 643
      DDP(I0)=DDO(J)
      I0=I0+1
643   CONTINUE
      DO 645 I=1,NX
645   AZE(I,I)=1+DDP(I)
647   DO 653 I=1,NX
      I1=I+1
      IF(NNX.LT.I1)  GO TO 113
      DO 649 J=I1,NNX
649   AZE(J,I)=AZE(I,J)
113   I2=I-1
      IF(I2.LT.1)  GO TO 653
      DO 653 J=1,I2
651   AW(J)=AZE(I,J)
      I3=J-1
      IF(I3.LT.1)  GO TO 115
      DO 652 KK=1,I3
652   AW(J)=AW(J)-AW(KK)*AZE(J,KK)
115   AZE(I,J)=AW((J)/AZE(J,J)
      AZE(I,I)=AZE(I,I)-AW(J)*AZE(I,J)
```

```
653  CONTINUE
     IF(KI.NE.5)  GO TO 659
     WRITE(6,655)
655  FORMAT(//,5X,5HL-AZE)
     DO 656 I=1,NNX
656  WRITE(6,657)  (AZE(I,J),J=1,NNX)
657  FORMAT(12F10.3)
659  DO 661 I=1,NX
     I1=I-1
     DO 661 KK=1,I1
661  AZE(NNX,I)=AZE(NNX,I)-AZE(I,KK)*AZE(NN-
        X, KK)
     DO 663 J=1,NX,1
     I=1+NX-J
     AZE(NNX,I)=AZE(NNX,I)/AZE(I,I)
     I2=I+1
     DO 663 KK=I2,NX
663  AZE(NNX,I)=AZE(NNX,I)-AZE(KK,I)*AZE(NN
        X,KK)
     DO 665 I=1,NX
     AW(I)=AZE(NNX,I)/R1(I)
665  X(I)=AW(I)+XO(I)
     IF(KI.NE.5)  GO TO 674
     WRITE(6,667)
667  FORMAT(//,5X,2HAW)
     WRITE(6,669)  (AW(I),I=1,NX)
669  FORMAT(12F10.3)
     WRITE(6,671)
671  FORMAT(//,5X,2HXO  )
     WRITE(6,669)  (XO(I),I=1,NX)
     WRITE(6,673)
673  FORMAT(//,5X,1HX)
```

```
      WRITE(6,669)(X(I),I=1,NX)
674   I=1
      DO 683 J=1,MAC
      IF(IX(K0,J+1).NE.1) GO TO 683
      AC(K0,J)=X(I)
      I=I+1
683   CONTINUE
      I0=0
      DO 681 I=2,MAX
      IF(IX(K0,I).NE.1)  GO TO 681
      I0=I0+1
      IF(AC(K0,I-1).GT.AY1(K0,I-1).AND.AC(K0.I
         -1).LT.AY2(K0,I-1)
         .AND.AC(K0,6).GT.AC.(K0,6))  GO TO 681
      AC(K0,I-1)=XO(I0)
681   CONTINUE
      CALL OUTZ
684   CALL ZYFT(FTO,PX,IM     )
      CALL QF(FTO,F,DPT,PT IM)
      IF(KI.NE.5)GO TO 689
      WRITE(6,685)
685   FORMAT(//,5X,6HFT0 * 2 * )
      WRITE(6,687)  (FTO(I),I=1,IM)
687   FORMAT(12F10.3)
689   IF(F.GT.FO)  GO TO 697
      FO=F
      DO 691 I=1,NX
691   XO(I)=X(I)
      I=1
      DO 692 J=1,MAC
      IF(IX(K0,J+1).NE.1) GO TO 692
      AC(K0,J)=XO(I)
```

```
      I=I+1
692   CONTINUE
      IF(KI.NE.5)  GO TO 695
      WRITE(6,693)  S
693   FORMAT(//,5X,1HS,F10.5)
695   IF(S.GT.0.1.AND.IDED.LT.KSP)  GO TO 6002
      IF(S.LE.0.1.OR.IDED.GE.KSP  (GO TO 6004
697   AD=ABS(AW(1))
      DO 699 I=1,NX
      BD=ABS(AW(I))
      IF(BD.GT.AD) AD=BD
699   CONTINUE
      IF(AD.GE.0.01)  GO TO 700
      WRITE(6,698)
698   FORMAT(//,5X,21H* * INCREMENT.LT.0.01 * *)
      GO TO 6004
700   DO 771 J=1,MAC
771   DDP(I)=DDP(I)*1.5
      IDED=IDED+1
      IF(IDED.GE.KSP)  GO TO 6004
      GO TO 645
6004  AS=SQRT(F)/IM
      BS=ABS(DPT(1))
      JC=1
      DO 775 I=2,IM
      DP=ABS(DTP(I))
      IF(DP.LE.BS)  GO TO 775
      BS=ABS(DPT(I))
      JC=I
775   CONTINUE
      BS=DPT(JC)
      D=DPT(JC)/PT(JC)
```

```
C
      WSITE(6,777)  KO
 777  FORMAT(//,5X,2H * *,3HNO,I2,2X,5HLAYER,2H **)
      IF(LF.EQ.1.AND.A0(K0,2).EQ.1.0)AC(KO,3)=AC
         (K0,1)
      IF(LF.EQ.1.AND.A0(K0,2).EQ.0.1)  AC(KO,3)=AC
         (K0,1)
      IF(IO.EQ.0)  GO TO 783
      DO 221 J=1,5
 221  AC1(K0,J)=AC(K0,J)
      AC1(K0,6)=AC(K0,6)-AC(K,Q7)*SH1
      AC1(K0,7)=AC(K0,7)
      DO 225 I=1,MTAN
      AC1(K0,3*I+5)=AC(K0, 3*I+5)
      IF(AC1(K0,3*I+5).EQ.0.0)GO TO 223
      AC1(K0,3*I+6)=AC(K0,3*I+6)+SH1
      GO TO 225
 223  AC1(K0,3*I+6)=AC(K0,3*I+6)
 225  AC1(K0,3*I+7)=AC(K0,3*I+7)
      WRITE(6,779)
 779  FORMAT(//,5X,5X,3HAC1)
      WRITE(6,781))  AC1(K0,I),I=1,MAC)
 781  FORMAT(12F10.3)
      GO TO 785
 783  WRITE(6,784)
 784  FORMAT(//,5X,2HAC)
      WRITE(6,781)  (AC  (K0,I),I=1,MAC)
 785  WRITE(6,787)
 787  FORMAT(/,11X,2HPX,8X,2HPT,7X3HPTO,7X,3HD-
         PT,8X,2HPX,8X,2HPT,
         7X,3HPTO,7X,3HDPT,8X,2HPX,8X,2HPT,7X,
         3HPTO,7X,3HDPT/)
```

```
      WRITE(6,789) (PX(I),PT(I),FTO(I),DPT(I),I=1,IM)
789   FORMAT(5X,12F10.3)
      WRITE(6,791) IDED,AS,JC,BS
791   FORMAT(//,5X,4HDEDI,     I3,5X,   2HAS,F7.3,5X,
       3HMAX,  I3,5X,2HBS,F7.3)
      IF(KFZ.NE.1) GO TO   793
      CALL PLOT(0.0,0.0,  3)
      CALL NEWPEN(2)
      CALL FIGT1(PX,PT,IM,K0)
      CALL PLOT(0.0,0.0,   3)
      CALL PLOT(0.0,0.0,   3)
      CALL NEWPEN(1)
      CALL FIGT1(PX,FTO,IM,K0)
      CALL PLOT(0.0,0.0,   3)
793   CONTINUE
      RETURN
      END
```

# 参考文献

## 第一、二、三章

王德人，非线性方程组解法与最优化方法，人民教育出版社，1979。

冯康等，数值计算方法，国防工业出版社，1978。

朱介寿，左海燕，约束条件下地球物理反演问题的数值解法，地球物理学报，25，582—593，1982。

何宝侃等，地球物理反问题中的最优化方法，地质出版社，1980。

周照襄等，解释重磁资料中的最优化选择法的若干问题，地球物理学报，23，1980。

杨篪引等，电子计算机应用数学，冶金工业出版社，1979。

南京大学，光学系统自动设计中的数值方法，国防工业出版社，1976。

南京大学，最优化方法，科学出版社，1978。

Avrid, M. 著，李元熹等译，非线性规划——分析与方法，上海科技出版社，1979。

Luenberger, D. G. 著，夏尊铨等译，线性与非线性规划引论，科学出版社，1980。

Aki, K., Christoffersson, A., Husebye, E. S., 1976. Three dimensional seismic structure of the lithosphere under Montana LASA, Bull. Seism. Soc. Am., 66, 501—524.

Aki, K., 1977, Three dimensional seismic velocity anomalies in the lithosphere, J. Geophys., 43, 235—242.

Andersen, R. S., Worthington, M. H., Cleary. J. R., 1972, Density modelling by Monte Carlo inversion—1. Methodology, Geophys. J. R. Astron. Soc., 29, 433—444.

Backus, G., Gilbert, F., 1976, Numerical applicationof a formalism for geophysical inverse problems, Geophys. J. Roy. Astron. Soc., 13, 247—276.

Backus, G., Gilbert, F., 1968, The resolving power

of gross earth data, Geophys. J. Roy. Astron. Soc., 16, 169—205.

Backus, G., Gilbert, F., 1970, Uniqueness in the inversion of inaccurate gross earth data, Phil. Trans. R. Soc. London Ser. A226, 123—192.

Bhattacharyya, B. K., Navolio, M. E., 1976, A fast Fourier transform method for rapid computation of gravity and magnetic anomalies due to arbitrary bodies, Geophys. Prospect., 24, 633—649.

Bhattacharyya, B. K., 1978, Computer modelling in gravity and magnetic interpretation, Geophys. 43, 912—929.

Borsting Pedersen, L., 1977, Interpretation of potential field data-A GeneraliZed inverse approach, Geophys. Prospect., 25, 109—230.

Braile, L. W.; 1973, Inversion of crustal seismic refraction and reflection data, J. Geophys. Res., 78, 7738—7744.

Braile, L. W., Keller, G. R., Peeples, W. J., 1974, Inversion of gravity data for two-dimensional density distributions, J. Geophys. Res., 79, 2017—2021.

Braile, L. W., Keller, G. R., 1975, Fine structure of the crust inferred from linear inversion of Rayleigh-wave dispersion, Bull. Seismo. Soc. Amer., 65, 71—83.

Burkhard, N., Jackson, D. D., 1976, Applications of stabilized linear inverse theory to gravity data, J. Geophys. Res., 81, 1513—1518.

Corbato, C. E., 1965, A least-squares procedure for gravity interpretation, Geophys., 30, 228—233.

Crosson, R. S., 1976, Crustal modelling of earthquake data, 1. simultaneous least squares estimation of hy-

pocenter and velocity parameters, J. Geophys. Res., 81, 3036—3046.

Der, Z., Masse, R., Landisman, M., 1970, Resolution of surface waves at intermediate distances, J. Geophys. Res., 75, 3399—3409.

Dunford, N., Schwartz, J., 1958, Linear operators—I, Interscience, New York.

Fisher, N. J., Howard, L. E., 1980, Gravity interpretation with the aid of quadratic programming. Geophys., 45, 403—419.

Green, W. R., 1975, Inversion of gravity profiles by use of a Backus Gilbert approach, Geophys., 40, 763—772.

Gilbert, F., 1971, Ranking and Windowing gross earth data for inversion and resolution, Geophys. J. R. Astron. Soc., 23, 125—128.

Inmann, J. R., Ryu. J., Ward, S. H., 1973, Resistivity inversion, Geophys., 38, 1088—1108.

Inmann, J. R., 1975, Resistivity inversion with ridge regression, Geophys., 40, 798—817.

Jackson, D. D., 1973, Marginal solutions to quasilinear inverse problems in geophysics, The edgehog method, Geophys. J. R. Astr., 35, 121—136.

Jackson, D. D., 1976, Most squares inversion, J. Geophys. Res., 81, 1027—1030.

Jackson, D. D., 1979, The use of a priori data to resolve nonuniqueness in Linear inversion, Geophys. J. Roy. Astron. Soc., 57, 137—157.

Johnson, W. W., 1969, A least squares method of interpreting magnetic anomalies caused by two—dimensional structures, Geophys., 34, 65—74.

Johnson, I. E., Gilbert, F., 1972, Inversion and

interference of teleseismic ray data, "Methods in computational physics", Vol. 12, Academic Press, New York, 231—266.

Jupp, D. , Vozoff, K. , 1975, Stable interactive method for the inversion of geophysical data, Geophys. Roy. Astron. Soc. , 42, 957—976.

Kennett, B.,Nolet, G.,1978, Resolution analysis for discrete systems, Geophys.J.Roy. Astron., 53,373—391.

Knopoff, L. ,1972, Observation and inversion of surface—wave dispersion, The upper mantle, Tectonophys. , 13, 497—519.

Kovach, R. L. , 1978, Seismic surface waves and crustal and upper mantle structure, Rev. Geophys. and Space physics, 16, 1—13.

Lanczos, C. ,1961,Linear differential operators, D. Van Nostrand Co. , London.

Lawson, C. L. , Hanson. R. J. , 1974, Solving least squares problems, Prentice—Hall, Inc.

Levenberg, K. , 1944, A method for solution of certain non-linear problems in least squares, Quart. Appl. Math. , 2, 164—168.

Luenberger, D. G. , 1965,Introduction to linear and non-linear programming,Addison—Wesley publishing Company.

Marquardt, D. W. , 1963, An algorithm for least—squares estimation of non—linear parameters, J. Soc. Indust. Appl. Math. , 11, 431—441.

McGrath, P. H. , Hood, P. J. , 1973, An automatic least-squares multimoder method for magnetic interpretation, Geophys. , 38, 1057—1071.

Murray, W. , 1972, Numerical method for uncons

trained optimization, Academic Press Inc. , New York.

Nagi, J. G ., Garde, S. C. , 1969, Symmetric matrix method for gravity interpretation, J. Geophsy. Res. , 74, 3804—3807.

Oldenburg, D. W. , 1974, The inversion and interpretation of gravity anomalies, Geophys. , 39, 526—536.

Parker, R. L. , 1970, The inverse problem of electrical conductivity in the mantle, Geophys. J. R. Astr. Soc. , 22, 121—138,

Parker, R. L. , 1975, Theory of ideal bodies for gravity interpretation, Geophys. J. R. Astr. Soc. , 42, 315—334.

Penrose, R. , 1955, A generalized inverse for matrics, Proc. Camb. Phil. Soc. , 51, 406—413.

Press. F. , 1968, Earth models obtained by Monte Carlo inversion, J. Geophys. Res. , 73, 5223—5234.

Rao, C. R. , Mitra, S. K. , 1971, Generalized inverse of matrics and its applications, John Wiley and Sons Inc. , New York.

Sabatier, P. C. , 1977, Positivity constraints in linear inverse problem—1, General theory, I, Applications, Geophys, J. R. Astr. Soc. , 48, 415—469.

Safon, C. , Vasseur, G. , Cuer,M. , 1977, Some applications of linear programming to the inverse gravity problem, Geophys. , 40, 1215—1229.

Schwab, F. A. , Knopoff, L. , 1972, Fast surface wave and free mode computations, "Methods in computational physics", Vol. 11, Academic Press.

Smith, M. L. , Franklin, J. N. , 1969, Geophysical application of generalized inverse theory, J. Geophys. Res. , 74, 2783—2785.

Talwani, M. , et al. , 1959, A crustal section across

the Puerto Rico Trench, J. Geophys. Res., 94, 1545—1555.

Tanner, J. G., 1967, An automated method of gravity interpretation, Geophys. J. R. Astr. Soc., 13, 339—347.

Vigneresse, J. L., 1977, Linear inverse problem in gravity profile interpretation, J. Geophys., 43, 193—213.

Vigneresse, J. L., 1978, Damped and constrained least squares method with application to gravity interpretation, J. Geophys., 45, 17—28.

Wiggins, R. A., 1972, The general linear inverse problem:implications of surface waves and free oscilations on earth structure, Rev. Geophys. and Space Phys., 10, 251—285.

Wollard, G. P., 1959, Crustal structure from gravity and seismic measurements, J. Geophys. Res., 64, 10, 1521—1543.

Wilds, D., Beightler, C., 1967, Foundations of optimization, Prentice Hall, Englewood Cliffs, N. J., 1967.

## 第四章

张之立、李钦祖等，唐山地震的破裂过程及其力学分析，地震学报，2，111—129，1980.

Hodgson, J, H., etc., 1965, Computer determined P—nodal solutions for the large earthquake of 1959—1962, Publ. Dom. Obs., Ottawa, 31, 123--143.

Knopoff, L., 1961, Analytical calculation of the fault-plane problem, pnbl. Dom. Obs., Ottawa, 24, 309—315.

Kasahara, I., 1963, Computer program of a fault plane solution, Bull. Seism. Soc. Amer., 53, 1—14.

Stevens, A. E., etc., 1968, A study of p—nodal solutions(1922—1962) in the Wicken s—Hodgson Catalogue, Bull. Seism. Soc. Amer., 58, 1071—1108.

Wickens, A. J. Hodgson. J. H., 1967, Computer reevaluation of earthquake mechanism solutions, Publ. Dom. Obs., Ottawa, 31, 1—560.

## 第五章

朱介寿、阎志德，一个测定近震震源参数的计算机程序，地震研究，4，1981.

唐国兴，用计算机确定地震参数的一个通用方法，地震学报，1，1979.

Flin, E. A., 1960, Local earthquake with an electronic computer, Bull. Seism. Soc. Am., 50, 467—470.

Geiger, K., 1910, Herdbestimmungen bei Eadbeben aus den Aukun—ftzeiten, K. Ges. Wiss. Gottingen, 4, 331—349.

Herrin, E. and Taggart, J., 1968, Source bias in epicenter determination, Bull. Seism. Soc. Am., 58, 1791—1796.

Lee, W. H. K. and Lahr, C., 1972, Hypo 71:Computer program for determining hypocenter, magnitude and first motion pattern of local earthquakes, U. S. Geol. Survey Open FileReport.

Lomnitz, C., 1977, A fast epicenter location program, Bull. Seism Soc. Am., 67, 425—431.

## 第六章

丁韫玉，用体波走时反演地壳结构，地球物理学报，36，1983.

Crosson, R. S., 1976, Crustal structure modelling of earthquake data, J. Geophys. Res., 81, 3036—3054.

Gangi, A. F., Yang, S. J., 1976, Travelling Curves

for reflections in dipping layers, Geophys. , 41, 425—440.

Johnson, S. H. , 1976, Interpretation of split-spread refraction data in term of plane dipping layers. Geophys. , 41, 418—424.

Ocola, L. A. , 1972, An algorithm for computation of source—to detector ranges from the travel time of multiply reflected sound wave, Geophys. , 37, 20—28.

## 第七章

B, 古登堡著，王子昌译，地球内部物理学，科学出版社，1965。

C. B. 奥菲塞著，王子昌等译，理论地球物理学引论，科学出版社，980。

宋仲和，朱介寿等，北京—萨哈林剖面的地幔纵向速度结构，地球物理学报，24，1981。

傅淑芳、刘宝诚、李文艺编，地震学教程，地震出版社，1981。

赵珠，华北地区上地幔速度结构，地球物理学报，26，1983。

Bates, A. C. , Kanasewich, E. R. , 1979, Inversion of seismic travel time using the Tau method, Geophys. J. R. Astr. Soc. , 47, 59—72.

Bruce, R. J. , Anderson, D. L. , 1968, Travel times, apparent velocities and amplitudes of body waves, Bull. Seis. Soc. Amer. , 58, 339—366.

Cleary, J. , Hales, A. , 1966, An analysis of the travel time of P waves to North American stations, in the distance range 32° to 100°, Bull. Seis. Sos. Amer. , 56, 467—489.

Gerver, M. L. ,Markushevich, V. M. ,1966, Determination of seismic wave velocity from the travel—time curve, Geophys. J. R. Astr. Soc. , 11, 165—173.

Gerver, M. L. , Markushevich, V. M. , Determining seismic wave velocity from travel—time curves, in "Computational Seismology", 1972.

Green, R. W. E. , Halis, A. L. , 1968, The travel times of P waves to 30°in the Central United States and upper mantle structure, Bull. Seis. Amer. , 58, 267—289.

Johnson, L. E. , Gilbert, F. , 1972, Inversion and interference for teleseismic ray data,"Methods in computational physics", Vol. 12. Academic press.

McMechan, G. A. , 1979, Low—velocity zone inversion by the Wiechert—Herglotz integral, Bull. Seis. Soc. Amer. , 69, 379—385.

## 第八章

冯锐、朱介寿等，利用地震面波研究中国地壳结构，地震学报，3，1981。

李白基等，地震面波频散的数字计算——方法与实验，地球物理学报，20，1977。

Braile, L. W. and Keller, G. R. , 1975, Fine structure of the crust inferred from linear inversion of Rayleigh—Wave dispersion, Bull. Seis. Soc. Am. , 65, 71—83.

Dunkin, J. W. , 1965, Computation of modal solutions in layered, elastic media at high frequencies, Bull. Seis. Soc, Am. , 55, 335—358.

Dziewonski. A. M. and Hales. A. L. , 1972, Numerical analysis of dispersed seismic waves, Seismology: Surface waves and earth oscillations, "Methods in Computational Physics", Vol. 11, Academic Press.

Haskell, N. A. , 1953, The dispersion of surface waves on multilayered media, Bull. Seis. Soc. Am. , 43, 17—34.

Knopoff, L. , 1964, A matrix method for elastic wave problem, Bull. Seis. Soc. Am. , 54, 431—438.

Kovach R. L. , 1978, Seismic surface waves and crustal and upper mantle structure, Reviews of Geophysics and Space Physics, 16, 1—13.

Satô, Y. , 1956, Analysis of dispersed surface waves by means of Fourier transform(Ⅱ), Bull. Earthq. Res. Inst. , 34, 131—138.

Schwab, F. and Knopoff, L. , 1970, Surface wave dispersions, Bull. Seis. Soc. Am. , 60, 321—344.

Schwab, F. and Knopoff, L. , 1972, Fast surface wave and free mode computation, "Methods in computational physics", Vol. 11, Academic Press.

Watson, T. H.,1970, A note on fast computation of Rayleigh wave dispersion in the multilayered elastic Half—space, Bull. Seis. Soc. Am. , 60, 161—166.

## 第九章

Andrianova, Z. S. , Keilis—Borok, V. I. , Levshin, A. L. , and Neigagz, M. G. , 1967, Seismic Love Wavev. New York: Consultants Bureau.

Bakun, W. H. and Eisenberg, A. , 1970, Fourier integrals and quadrature—introduced aliasing, Bull. Seism. Soc. Am. , 60, 1291—1296.

Cagniard, L. 1939, Reflexion et refraction des ondes seismiques progressives, Gauthier—Villars, Paris.

červený, V. , 1961,The amplitude curves of the reflected harmonic waves around the critical point. Studia Geophys. Geod. , 5, 319—351.

Chapman, C. H. 1973, The earth flattening transformation in body wave theory, Geophys. J. R. Astr. Soc. 35, 55—70.

Chapman, C. H. , 1978, A new method for compting synthetic seismograms, Geophys. J. R. Astr. Soc. , 54,

481—518.

Cormier, V. F. and Richards, P. G.,1977, Full wave theory applied to a discontinuos velocity increace: the inner core boundary, J. Geophys. , 43, 3—31.

Dukin, L. W.,1965, Computation of modal solution in layered elastic media at high frequencies, Bull. Seism. Am. , 55, 335—358.

Fuchs, K. , 1968, Das reflexions und transmissions-vermögen einesgeschichteten mediums mit beliebiger tief-enverteilung der elastischen modula und der dichte für schrögen einfall ebener wellen, Z. Geophys.,34, 389—413.

Gilbert, F. and Helmberger, D. V. , 1972, Generalized ray theory for a layered sphere,Geophys. J. R. Astr. Soc. , 27, 57—80.

Kennett, B. L. N. ,1974, Reflection rays and reverberations, Bull. Seism. Soc. Am. , 64, 1665—1696.

Kennett, B. L. N. , 1975, The effect of attenuation on seismograms. Bull. Seism. Soc. Am. , 65, 1643—1651.

Mülel, G. 1968, Theoretical seismograms for some type of pointsource in layered media, a. Part 1: Theory, Z. Geophys. , 34, 15—35. , b. Part 2: Numerical calculation, Z. Geophys. , 34, 147—162.

Wiggins R. A. and Helmberger, D. V. , 1974, Synthetic seismograms computation by expansion in generalized rays, Geophys. J. R. Astr. Soc. , 37, 73—90.

## 第十章

Cervenÿ, V., Ravindra, R., 1971, Theory of seismic head waves.

Gebrande, H., 1976, A Seismic—ray tracing method for two—dimen—sional inhomogeneous media, in“Explo-

sion seismology in Certral Europe".

Grubbe, K., 1976, Seismic-refraction measurements along two crossing profiles in northern Germany and their interpretation by a ray-tracing method, in "Explosion Seismology in Central Europe".

Hiller, H., Gebrande, H., 1976, Crustal structure in southeastern Bavaria derived from seismic—refraction measurements by ray—tracing methods, in "Explosion Seismology in Central Europe".

Will, M., 1976, Calculation of travel times and ray paths for lateral inhomogeneous media, in "Explosion Seismology in Central Europe".